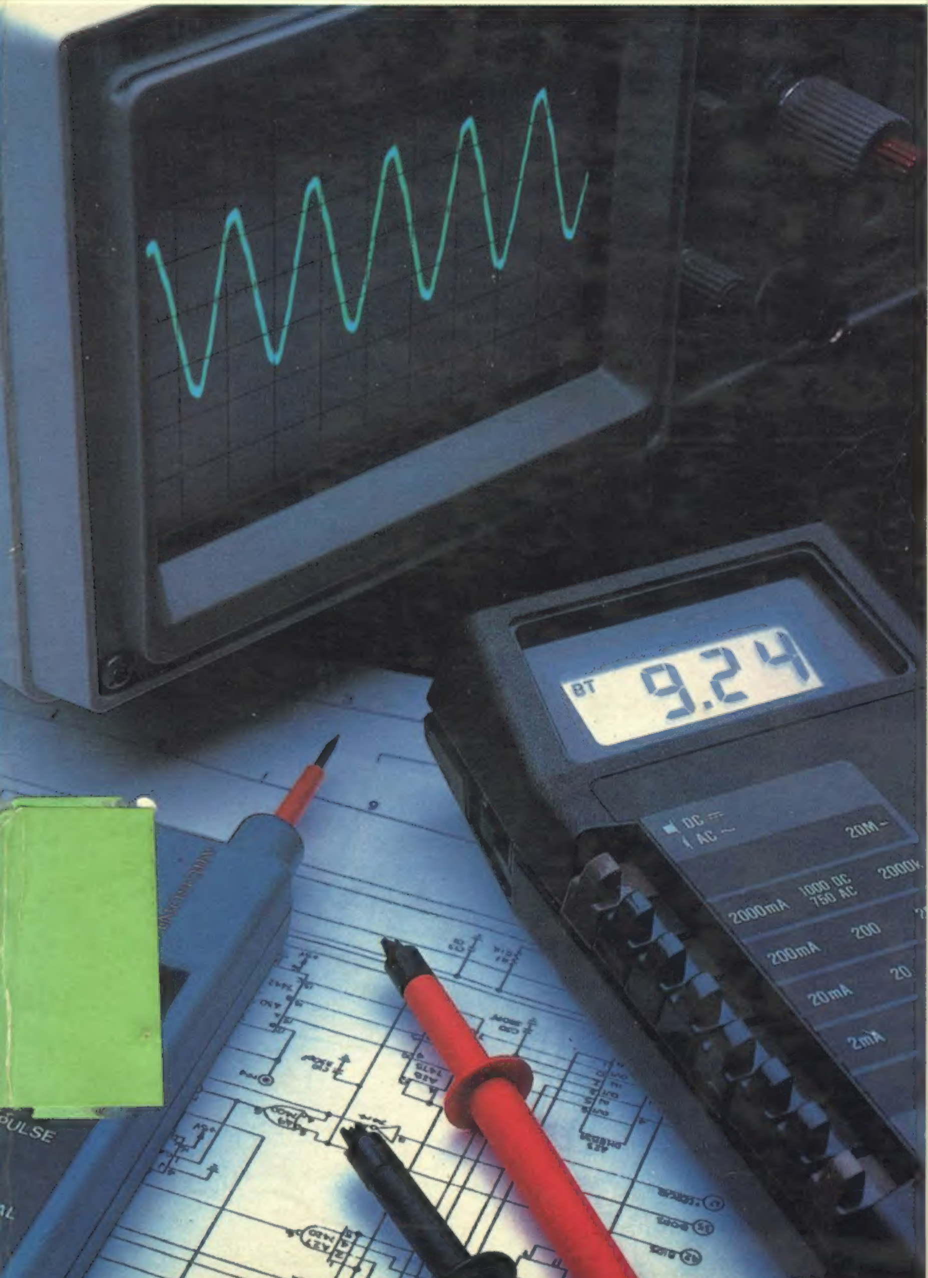
Collana Capire l'Elettronica e l'Informatica

Capire

10

l'Elettronica Digitale: Guasti e Manutenzione


BT 9.24
DC
AC
20M
2000k
2000mA
1000 DC 750 AC
200mA
200
20mA
20
2mA
PULSE

La manutenzione dei sistemi digitali, una necessità sempre crescente da quando i sistemi sono passati all'elettronica digitale. Ecco presentati in una forma di facile comprensione i concetti di base e le tecniche di ricerca dei guasti per avviarsi al lavoro di riparazione e manutenzione di apparati digitali. Completamente illustrato. Ideale per adattarsi alle diverse caratteristiche di apprendimento individuale.

Di:
Don L. Cannon

The Texas Instruments
Learning Center

tecniche nuove

Don L. Cannon

l'Elettronica Digitale: Guasti e Manutenzione

RESTITUIRÒ QUESTO LIBRO ENTRO IL

~~6 GIU. 1996~~	24 GIU. 2013	
~~28-3-00~~		
~~28-04-00~~(R)		
5-9-00		
~~6/10/00~~(R)		
~~29-12-00~~		
8/10/01		
1/07/02		
11/02/03		
15/03/03		
~~29/10/03~~		
27-06-04		
12/3/05		
26/10/11		

Don L. Cannon

l'Elettronica Digitale: Guasti e Manutenzione

X CIRCOSCRIZIONE - ROMA
BIBLIOTECA COMUNALE
VIA F. SERAFINI 55

Biblioteca Centro Culturale
RAFFAELLO
Via Tuscolana 1111
Circ.ne X - Comune di Roma

SEZIONE PRESTITO
GENERALE
N. 5843

PRESTITO
Collocazione N. 9250
621.381/CAN 2179

tecniche nuove

EDIZIONE ORIGINALE
Understanding Digital Troubleshooting - Don L. Cannon
© 1983, 1984, Texas Instruments Incorporated

EDIZIONE ITALIANA
© 1988 Tecniche Nuove, via Ciro Menotti 14, 20129 Milano

ISBN 88 7081 3177

Tutti i diritti sono riservati. Nessuna parte del libro può essere riprodotta o diffusa con un mezzo qualsiasi, fotocopie, microfilm o altro, senza il permesso scritto dell'editore.

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher.

Stampa: Polver, Milano

Scritto da:

Don L. Cannon, Ph. D.
Docente presso il Dipartimento di
Ingegneria Elettronica dell'Università
del Texas, Consulente del Centro Stampa
e Documentazione della Texas Instruments.

Con i contributi di:

Gerald Luecke, Manager per lo sviluppo dei
prodotti tecnici
Centro Stampa e Documentazione della
Texas Instruments

Charles W. Battle
Ken M. Krone
Les Mausir
Pete Sanborn
Il personale del Centro Stampa e Documentazione
della Texas Instruments.

Ringraziamenti a:

Mike Henderson per i suoi significativi contributi nell'isolare i guasti usando tecniche hardware e software; Tektronix Inc., Simpson Electric Co., John Fluke Mfg. Co. e Hewlett Packard Company per le fotografie della strumentazione.

Copertina:

l'illustrazione rappresenta la strumentazione di base per la manutenzione dei sistemi elettronici digitali: un oscilloscopio, una sonda logica, un voltmetro digitale con i contatti e uno schema elettrico di un circuito.

Indice

Prefazione VIII

Capitolo *Pagina*

1 *Fondamenti dei sistemi digitali* 1
Quiz 33
2 *Funzioni dei sistemi digitali* 35
Quiz 57
3 *Fondamenti di ricerca dei guasti* 59
Quiz 95
4 *Aspetti di logica combinatoria* 97
Quiz 118
5 *Problemi di logica sequenziale* 119
Quiz 151
6 *Aspetti delle memorie* 153
Quiz 176
7 *Aspetti di ingresso/uscita (I/O)* 179
Quiz 208
8 *Principali aspetti della temporizzazione* 209
Quiz 231
9 *Tecniche avanzate* 233
Quiz 267

Appendice 270
Glossario 275
Indice analitico 279
Soluzioni dei quiz 281

Prefazione

Poiché attraverso l'impiego di tecniche digitali vengono realizzati nuovi sistemi e riprogettati quelli ormai superati, si avverte il crescente bisogno di persone in grado di localizzare i guasti nei sistemi digitali. L'esperienza in questo campo è utile non solo per fornire assistenza in caso di guasti, ma anche per impostare la manutenzione preventiva.

Questo libro è stato scritto per coloro che desiderano o devono apprendere le tecniche impiegate in questa attività di assistenza che si sta sviluppando velocemente. Si rivolge sia a chi desidera effettuare riparazioni su macchine proprie per hobby, sia a chi intende intraprendere un'attività professionale in questo campo.

Il primo obiettivo è stato quello di spiegare le nozioni basilari dell'elettronica e i concetti ingegneristici (che sono argomenti prettamente tecnici) ad un livello ed in modo tale da renderli comprensibili al lettore medio interessato all'aspetto tecnico del problema; nella trattazione si è quindi fatto uso di nozioni matematiche poco più complesse rispetto alle quattro operazioni fondamentali di addizione, sottrazione, moltiplicazione e divisione.

Il secondo obiettivo è stato quello di mostrare alcuni esempi di applicazioni pratiche che utilizzano procedimenti logici ed intuitivi di sostituzione ed eliminazione, uniti a tecniche di misura, allo scopo di spiegare le tecniche usate e di rafforzare i concetti di base.

Per essere sicuri che i principi fondamentali delle tecniche digitali vengano acquisiti, nei primi due capitoli sono esaminati i fondamenti e le funzioni dei sistemi digitali, cui seguono i fondamenti delle tecniche di ricerca guasti. Dopo aver dato queste nozioni basilari, i rimanenti capitoli illustrano le tecniche di ricerca guasti che vengono utilizzate per trovare i guasti nei circuiti logici, nelle memorie e nei dispositivi di ingresso/uscita di un sistema digitale. Concludono il libro alcuni capitoli specifici concernenti la strumentazione più recente per localizzare gli errori di temporizzazione ed i guasti nei circuiti VLSI (Very large scale integration, integrazione su larghissima scala).

Tutti i libri della collana "Capire" cercano di facilitare la comprensione dei principi e dei concetti base dell'elettronica. Uno degli scopi di questo libro sarà stato raggiunto se il lettore avrà afferrato questi concetti così da sentirsi sempre più familiare con il mondo dell'elettronica. Un secondo e più importante risultato sarà stato raggiunto se il lettore sarà in grado, grazie a questo libro, di provvedere realmente all'eliminazione dei guasti in quest'era dominata dall'elettronica digitale.

D.L.C.

Fondamenti dei sistemi digitali

SCOPO DEL LIBRO

Nonostante la tecnologia dei semiconduttori e dei circuiti integrati abbia contribuito ad aumentare di molto l'affidabilità dei sistemi digitali, si presentano ancora occasioni in cui detti sistemi digitali non funzionano correttamente. In questo caso si dice che c'è un "bug" (baco) o un difetto. Il procedimento che permette di localizzarlo è detto eliminazione degli errori (troubleshooting) o "debugging". Per scoprire un guasto, un "bug" o un difetto in un sistema digitale occorre conoscere il funzionamento del sistema stesso; per capire il funzionamento del sistema dev'essere noto il funzionamento dei singoli componenti al suo interno.

Questo libro è stato scritto per aiutare a conoscere il funzionamento dei sistemi digitali e dei loro componenti, in modo tale da poter localizzare le parti difettose responsabili del malfunzionamento. L'intento specifico è quello di mettere in relazione il malfunzionamento di un sistema con il difettoso funzionamento di quel suo componente che lo ha causato. Poiché riuscire a ricondurre il guasto di un sistema ad un suo specifico sottosistema, ad un suo componente o ad una sua parte richiede misurazioni elettriche in diversi punti del sistema stesso, il funzionamento di un sistema o di un suo componente sarà rappresentato dal punto di vista delle tecniche di misura. Le informazioni acquisite attraverso le misure, integrate dalla conoscenza delle tecniche logiche di isolamento ed eliminazione, forniscono al lettore gli strumenti necessari per individuare e riparare i guasti nei moderni sistemi digitali. Inoltre, poiché deve essere noto il funzionamento dei sistemi digitali fondamentali, il libro fornisce al lettore un'adeguata rassegna di tali sistemi.

SCOPO DEL CAPITOLO

Il funzionamento di un sistema digitale si basa sulle proprietà fondamentali dei segnali elettrici, in quanto il sistema è formato da componenti digitali connessi fra di loro. Questo capitolo riguarda i principi fondamentali dei sistemi digitali. Esamina non solo le proprietà fondamentali dei segnali elettrici, ma anche il funzionamento di molti componenti digitali.

I segnali analogici e digitali sono tipi comuni di segnali elettrici. I segnali analogici variano con continuità e gradualmente nel tempo. I segnali digitali variano rapidamente tra livelli di tensione discreti.

SEGNALI ELETTRICI

Alla maggior parte dei lettori sarà noto che ci sono due tipi fondamentali di segnali elettrici: quelli analogici e quelli digitali. I segnali che hanno una variazione continua di ampiezza o frequenza sono detti segnali analogici. Un segnale analogico può essere costituito da variazioni di tensione continua (c.c.) o di tensione alternata (c.a.).

Il segnale digitale invece può assumere solitamente due livelli ben definiti di tensione ed ha solo occasionali e rapide transizioni tra questi due livelli.
Nella maggior parte dei sistemi, le variazioni dei valori o dei livelli del segnale nel tempo, per entrambi i tipi di segnale, sono in relazione con il contenuto d'informazione del sistema. Infatti, i segnali elettrici vengono usati principalmente per trasferire l'informazione da una parte all'altra di uno stesso sistema o da un sistema all'altro.

CIRCUITI IN TENSIONE CONTINUA (c.c.)

Gli strumenti di misurazione utilizzati per verificare il funzionamento dei circuiti consentono la lettura dei valori tramite: cambiamenti della posizione di un ago su di una scala (letture analogiche) o numeri discreti su un display digitale (letture digitali).

Un semplice circuito in tensione continua è mostrato nella *figura 1.1a.* Questo circuito è formato semplicemente da due resistenze collegate in serie e da un alimentatore. In figura sono mostrati anche due strumenti di misura fondamentali, l'amperometro ed il voltmetro digitale, che verranno spesso impiegati negli esempi trattati in questo libro. L'amperometro indica il valore della corrente elettrica misurata attraverso la posizione di un ago su di una scala graduata. È questo un amperometro analogico. Il voltmetro indica il valore della tensione elettrica con un numero sul suo display digitale. Si tratta di un voltmetro digitale. Nella misurazione di grandezze elettriche vengono impiegati sia strumenti analogici che strumenti digitali. Il circuito dell'esempio e gli strumenti indicati possono servire come base per esaminare le proprietà fondamentali dei circuiti elettrici.
L'alimentatore costituisce la fonte dell'energia elettrica e fornisce le correnti e le conseguenti cadute di tensione che si verificano nel circuito. La tensione elettrica è la forza che provoca il passaggio della corrente elettrica in un circuito. La corrente è il flusso degli elettroni, particelle cariche negativamente, in un materiale; conduttori metallici come l'oro, l'argento e il rame hanno un grande numero di elettroni liberi ed oppongono quindi poca resistenza al passaggio di corrente. Il carbonio ha un numero di elettroni liberi molto minore e, pertanto, oppone una resistenza più elevata; perciò molte resistenze, come quelle mostrate in *figura 1.1a*, sono fatte in carbonio.

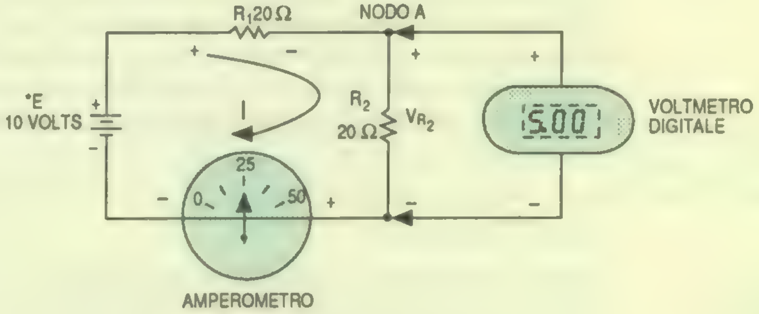
La legge di Ohm

La legge di Ohm afferma che la caduta di tensione ai capi di un resistore è proporzionale alla corrente che passa attraverso il resistore.

La relazione fondamentale che sussiste tra la corrente che percorre una resistenza e la tensione applicata ai suoi capi è mostrata nella *figura 1.1b.* Questa relazione nella quale la tensione applicata ai capi di una resistenza è uguale al prodotto di tale resistenza per la corrente che la percorre è nota come legge di Ohm. La resistenza è espressa in ohm, la corrente in ampere e la tensione in volt. La caduta di tensione ai capi di una resistenza è direttamente proporzionale alla corrente che la percorre. La resistenza dissipa potenza trasformando potenza elettrica in calore. La quantità di potenza dissipata da una resistenza può essere calcolata mediante una delle seguenti relazioni, dove la tensione è espressa in volt, la corrente in ampere e la resistenza in ohm:

1. La tensione applicata ai capi di una resistenza moltiplicata per la corrente che attraversa la resistenza stessa;

Figura 1.1. Fondamenti dei circuiti in corrente continua


R₁20 Ω
NODO A
*E
10 VOLTS
I
R₂
20 Ω
V_R₂
5.00
VOLTMETRO
DIGITALE
25
0
50
AMPEROMETRO

a. Semplice circuito in C.C.

Legge di Ohm per il circuito:

$$E = I_T R_T$$

Per un resistore nel circuito:

Legge di Ohm: $V_R = IR$

$$I = \frac{V_R}{R}$$

$$R = \frac{V_R}{I}$$

Potenza dissipata in un resistore: $P = V_R I$

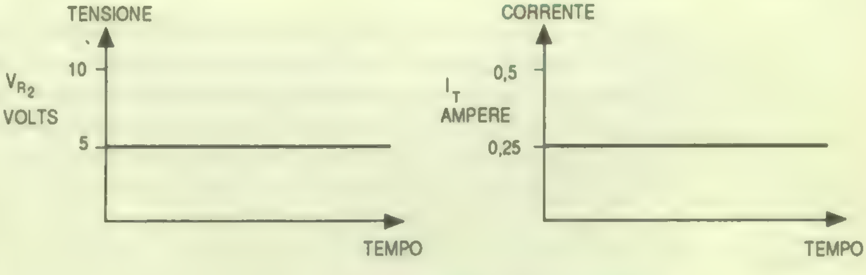
$$P = I^2 R$$

$$P = \frac{V_R^2}{R}$$

Dove: V è la caduta di tensione in volt
I è la corrente di ampere
R è la resistenza in ohm
P è la potenza in watt.

*Il simbolo E (che sta per forza elettromotrice) è usato per il generatore principale di potenza in un circuito, mentre V è usato per le cadute di tensione lungo il circuito.

b. Relazioni elettriche



c. Corrente e tensione in funzione del tempo

2. La resistenza moltiplicata per il quadrato della corrente che la attraversa;
3. Il quadrato della tensione ai capi della resistenza diviso per la resistenza stessa.

Nel circuito di *figura 1.1a* la forza elettromotrice E è costante; perciò anche la corrente è costante. Poiché queste grandezze assumono valori costanti le rappresentazioni delle tensioni e delle correnti nel circuito in funzione del tempo sono linee orizzontali, come mostrato in *figura 1.1c*. Queste tensioni e correnti continue sono importanti nella verifica del funzionamento di un circuito anche nei sistemi digitali ad alta frequenza.

La legge di Kirchhoff

La corrente uscente da un nodo di un circuito deve uguagliare la corrente entrante in esso (legge di Kirchhoff delle correnti). Inoltre, in un circuito chiuso, la somma delle cadute di tensione ai capi di tutti i componenti del circuito deve uguagliare la tensione dell'alimentatore (legge di Kirchhoff delle tensioni).

Allo scopo di determinare i valori della corrente e della tensione nel circuito mostrato in *figura 1.1a*, devono essere note alcune leggi fondamentali che regolano il funzionamento di un circuito. La prima di queste regole afferma che la somma delle correnti entranti in una connessione in cui confluiscono due o più rami o componenti del circuito deve essere pari alla somma delle correnti uscenti dalla connessione stessa. Un punto di connessione è noto col nome di "nodo". Questa regola è conosciuta come legge di Kirchhoff delle correnti dal nome di Gustav R. Kirchhoff che per primo la formulò.

Esiste una regola simile, nota con il nome di legge di Kirchhoff delle tensioni, che può essere applicata alle tensioni elettriche. Essa afferma che la somma delle tensioni, misurate lungo i rami di un circuito tali da formare un percorso chiuso, o "maglia", deve essere nulla. In altre parole ciò significa che la forza elettromotrice applicata ad un circuito dev'essere uguale alla somma delle cadute di tensione che si verificano su tutti i componenti del circuito stesso.

Esempio

In un circuito in cui tutti i componenti sono collegati in serie passa la stessa corrente. Moltiplicando questa corrente per la resistenza totale si ottiene la caduta di tensione totale del circuito.

In *figura 1.1a* il percorso chiuso è formato dall'alimentatore E e dalle resistenze R_1 e R_2. La somma delle cadute di tensione ai capi di R_1 e R_2, indicate come V_{R1} e V_{R2}, deve essere uguale al valore della forza elettromotrice E. Inoltre, se si suppone nulla la corrente che passa attraverso il voltmetro, la corrente che giunge al nodo A da R_1 uscirà dal nodo stesso passando attraverso R_2 e tornerà all'alimentatore E attraversando l'amperometro. In altre parole, se si può supporre nulla la corrente attraverso il voltmetro, tutti gli elementi del circuito formato da E, R_1 ed R_2 e collegati in serie sono percorsi dalla stessa corrente. Se chiamiamo I l'intensità di questa corrente, la caduta di tensione ai capi di R_1 è IR_1, mentre quella ai capi di R_2 è IR_2; perciò la caduta di tensione totale attraverso R_1 ed R_2 è $IR_1 + IR_2$ In altri termini, la caduta di tensione ai capi di R_1 e R_2 è $I(R_1 + R_2)$; cioè è pari a I volte la somma di R_1 e R_2 Questa tensione deve essere uguale alla forza elettromotrice E; perciò $E = I(R_1 + R_2)$. Poiché $I = E/R_T$ (dove R_T è la resistenza totale del circuito), la corrente che percorre il circuito si trova come:

$$I = 10V/(20\Omega + 20\Omega)$$
$$I = 10V/40\Omega$$

$$I = 0{,}25 \text{ ampere}$$

Una volta nota l'intensità della corrente, la tensione applicata ai capi della resistenza R_2 può essere calcolata come:

$$V = IR_2$$
$$V = 0{,}25A \times 20\Omega$$
$$V = 5 \text{ volt}$$

Il voltmetro indica 5 volt perché sta misurando la caduta di tensione ai capi della resistenza R_2 Inoltre, poiché anche R_1 è di 2O ohm, la caduta di tensione ai capi di R_1 è di 5 volt e la somma delle cadute di tensione equivale alla tensione dell'alimentazione.

CIRCUITI IN TENSIONE ALTERNATA (c.a.)

I circuiti in tensione alternata sono percorsi da correnti e tensioni la cui polarità (positiva o negativa) varia in funzione del tempo. La più comune tensione alternata è quella fornita dalla rete elettrica domestica. Questa tensione, espressa in funzione del tempo, ha una rappresentazione a forma d'onda come mostrato nella *figura 1.2b*. Questo tipo di curva è nota come sinusoide. Il circuito utilizzato nella *figura 1.1a* è riportato anche in *figura 1.2a*, ma in questo caso la forza elettromotrice è sinusoidale e gli strumenti usati sono adatti alla misura di correnti e tensioni alternate. Infatti gli strumenti per misurare le tensioni continue non sono utilizzabili per tensioni alternate. Per misurare correnti e tensioni alternate occorre utilizzare strumenti per corrente alternata.

Tensione

Il valore RMS (Media quadratica) si ottiene facendo la radice quadrata della media di un certo numero di valori istantanei di una forma d'onda elevati al quadrato.

La tensione e la corrente alternata possono essere caratterizzate da una qualsiasi delle tre grandezze riportate in *figura 1.2b*. Il valore di picco è misurato dal livello di riferimento (lo zero) al picco positivo o negativo del segnale. Il valore "da picco a picco" è misurato dal picco positivo al picco negativo. Per onde sinusoidali, il valore comunemente usato è la media quadratica (o, brevemente, l'rms, da root-mean-square): è il valore che produce la medesima dissipazione di potenza in una resistenza, qualora la si alimentasse con tensione continua dello stesso valore.

Il valore rms (valore efficace) è un valore calcolato. Come mostrato in *figura 1.2c*, vengono rilevati diversi valori istantanei della sinusoide in corrispondenza di alcuni punti equidistanti durante il semiperiodo positivo della sinusoide, e ciascuno di questi valori viene elevato al quadrato e sommato ai quadrati degli altri valori. La somma di questi quadrati viene divisa per il numero dei valori considerati, al fine di ottenere una media. Infine, si estrae la radice quadrata di tale valore ottenendo così l'rms del segnale. Per un'onda sinusoidale il valore rms è pari a 0,707 volte il valore di picco. La relazione tra rms e valore di picco è rappresentata in *figura 1.2b*.

Le due proprietà di un'onda periodica come la sinusoide sono il periodo e la frequenza. Quando su una forma d'onda si ha un punto con un valore già trovato precedentemente si dice che la forma d'onda ha compiuto un periodo. La frequenza è il numero dei periodi al secondo, detti hertz.

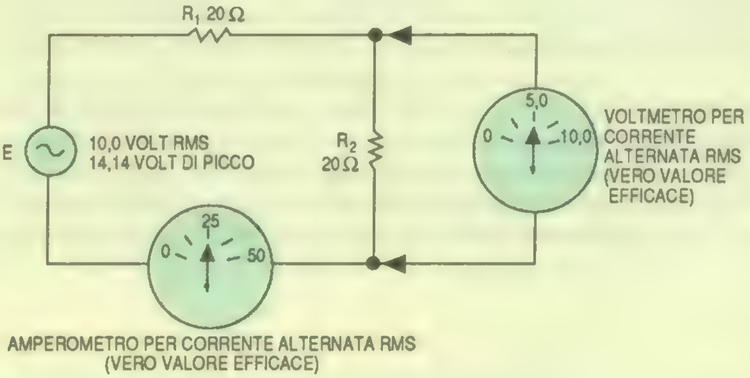
Ciclo

Un'altra importante proprietà della sinusoide è il ciclo. Esso si ripete un certo numero di volte al secondo. La grandezza che esprime quante volte al secondo si ripete un ciclo completo della sinusoide è detta frequenza e viene espressa in hertz, dove 1 hertz (Hz) è pari ad un ciclo al secondo. L'intervallo compreso tra due punti omologhi di un'onda è detto periodo, ed è indicato in *figura 1.2b* con il simbolo T. Il periodo è il reciproco della frequenza, è cioè 1 diviso per la frequenza. Pertanto, una sinusoide che ripeta il ciclo 100 volte al secondo ha una frequenza di 100 hertz ed un periodo di 1/100 di secondo oppure di 0,01 secondi. Questo periodo viene più propriamente indicato come 10 millesimi di secondo o 10 millisecondi.

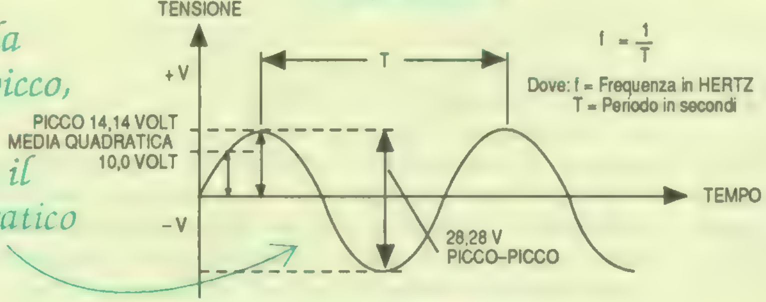
Fase

La scala orizzontale del grafico di un'onda sinusoidale può anche essere espressa in gradi. Un ciclo consta di 360 gradi. Con l'asse orizzontale suddivisa in gradi, la fa-

Figura 1.2. Fondamenti dei circuiti in corrente alternata


R$_1$ 20 Ω
E
10,0 VOLT RMS
14,14 VOLT DI PICCO
R$_2$
20 Ω
5,0
0
10,0
VOLTMETRO PER CORRENTE ALTERNATA RMS (VERO VALORE EFFICACE)
25
0
50
AMPEROMETRO PER CORRENTE ALTERNATA RMS (VERO VALORE EFFICACE)

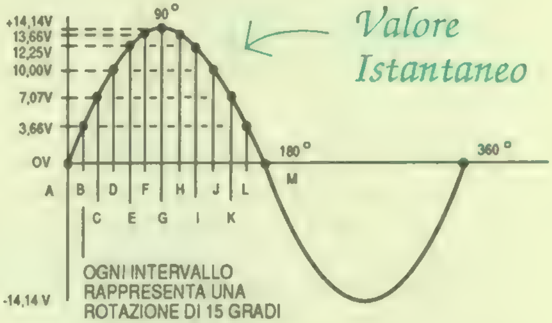
a. Semplice Circuito in AC


TENSIONE
+V
-V
T
f = 1/T
Dove: f = Frequenza in HERTZ
T = Periodo in secondi
PICCO 14,14 VOLT
MEDIA QUADRATICA
10,0 VOLT
TEMPO
28,28 V
PICCO-PICCO

b. Forma d'onda di Tensione in AC

Tre valori definiscono la tensione di picco, la tensione picco-picco e il valore quadratico medio della sinusoide

Figura 1.2. Fondamenti dei circuiti in corrente alternata (continua)



VALORI ISTANTANEI DI UN'ONDA DI TENSIONE SINUSOIDALE

Gradi di Rotazione	Punto	Valore Istantaneo di Tensione	Quadrato della Tensione Istantanea
0	A	0,0	0,0
15	B	3,66	13,396
30	C	7,07	49,985
45	D	10,00	100,000
60	E	12,25	150,063
75	F	13,66	186,596
90	G	14,14	199,939
105	H	13,66	186,596
120	I	12,25	150,063
135	J	10,00	100,000
150	K	7,07	49,985
165	L	3,66	13,396
180	M	0,0	0,0

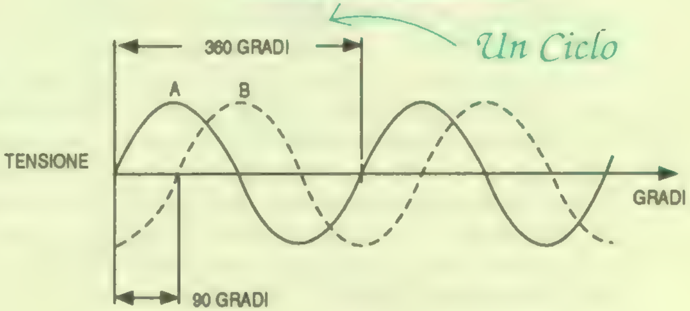
TOTALE = 1200,009

MEDIA = $\frac{1200}{12} = 100$

$V_{RMS} = \sqrt{100} = 10$ V

$10\ V_{RMS} + 14.14\ V_{PICCO} = 0.707$

c. Calcolo del valore quadratico medio



d. Fase

I confronti fra forme d'onda sinusoidali rappresentate graficamente in funzione del tempo forniscono uno strumento per misurare la fase delle forme d'onda.

se di un'onda può essere messa in relazione con un punto di riferimento dato. Inoltre l'onda di ingresso può essere confrontata con un'onda in uscita, generata dall'ingresso stesso, e il ritardo di fase tra le due può venire misurato. In figura 1.2d, l'onda B è in ritardo di 90 gradi rispetto all'onda A. (La frequenza di due onde deve essere esattamente la stessa per poterne confrontare le fasi.)

La legge di Ohm in tensione alternata

Mentre i valori picco-picco, di picco o RMS della corrente o della tensione possono essere utilizzati per calcolare valori del circuito, si deve usare molta attenzione nell'utilizzare le stesse unità di misura in ciascun calcolo.

Poiché il circuito in *figura 1.2a* contiene solo resistenze, si può applicare la legge di Ohm per calcolarne correnti e tensioni. Nei calcoli si possono usare indifferentemente il valore di picco, quello picco-picco oppure il valore quadratico medio (rms), ma è necessario mantenere sempre la stessa unità. La maggior parte dei voltmetri e degli amperometri per corrente alternata ha scale graduate secondo i valori rms. Per il circuito in *figura 1.2a* si ha:

$$I_T = E / R_T = 10/40 = 0{,}25 \text{ ampere rms}$$

$$V_{R2} = IR_2 = 0{,}25 \times 20 = 5 \text{ volt rms}$$

Di conseguenza l'amperometro indica 0,25 ampere e il voltmetro indica 5 volt. Anche in questo caso si suppone che attraverso il voltmetro non passi corrente.

SEGNALI COMPLESSI

Segnali analogici

Un segnale analogico può essere una combinazione di molte e differenti forme d'onda sinusoidali.

Se si vuole che una tensione, continua o alternata, trasporti un'informazione, qualche parte del segnale deve essere correlata all'informazione stessa. Per esempio, 10 differenti valori di tensione continua possono rappresentare 10 numeri differenti. 10 diverse frequenze di tensione sinusoidale possono servire allo stesso scopo. Analogamente, un gran numero di sinusoidi di diversa frequenza e ampiezza può essere sommato per generare l'onda del segnale desiderato. L'onda mostrata in *figura 1.3* rappresenta un segnale di questo tipo. Questo è un segnale analogico in quanto la sua ampiezza varia da un valore all'altro in modo continuo in funzione del tempo. Diverse combinazioni delle sinusoidi che lo compongono generano onde diverse. Quando le numerose e diverse sinusoidi prodotte dagli strumenti di un'orchestra e dalle voci di un coro sono raccolte da microfoni e convertite in segnali elettrici che vengono sommati tra loro, l'onda risultante ha una forma molto complicata ed è variabile con continuità. Questi segnali possono essere inviati su lunghe distanze usando linee telefoniche o segnali radio e possono essere riprodotti sotto forma di suoni nel luogo in cui vengono ricevuti.

Segnali digitali

La stessa informazione può essere rappresentata da segnali digitali se vengono usati alcuni metodi di codifica. (Le codifiche digitali saranno trattate più avanti.) Anche

Figura 1.3.
Segnale analogico non sinusoidale

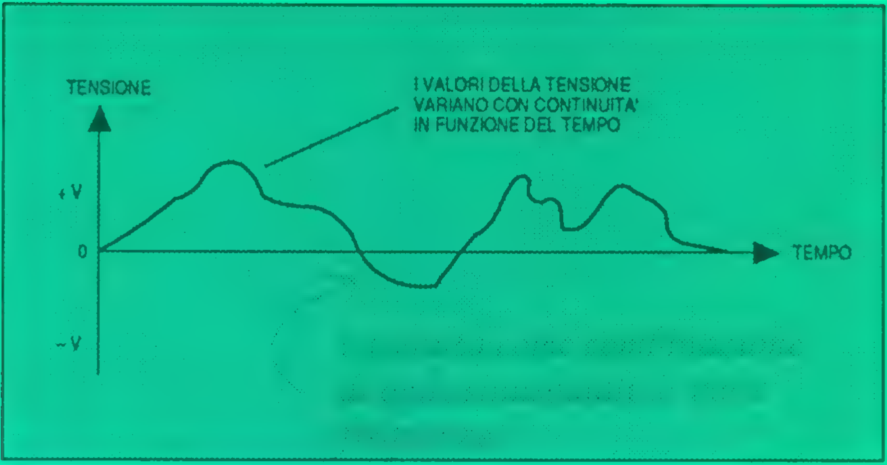
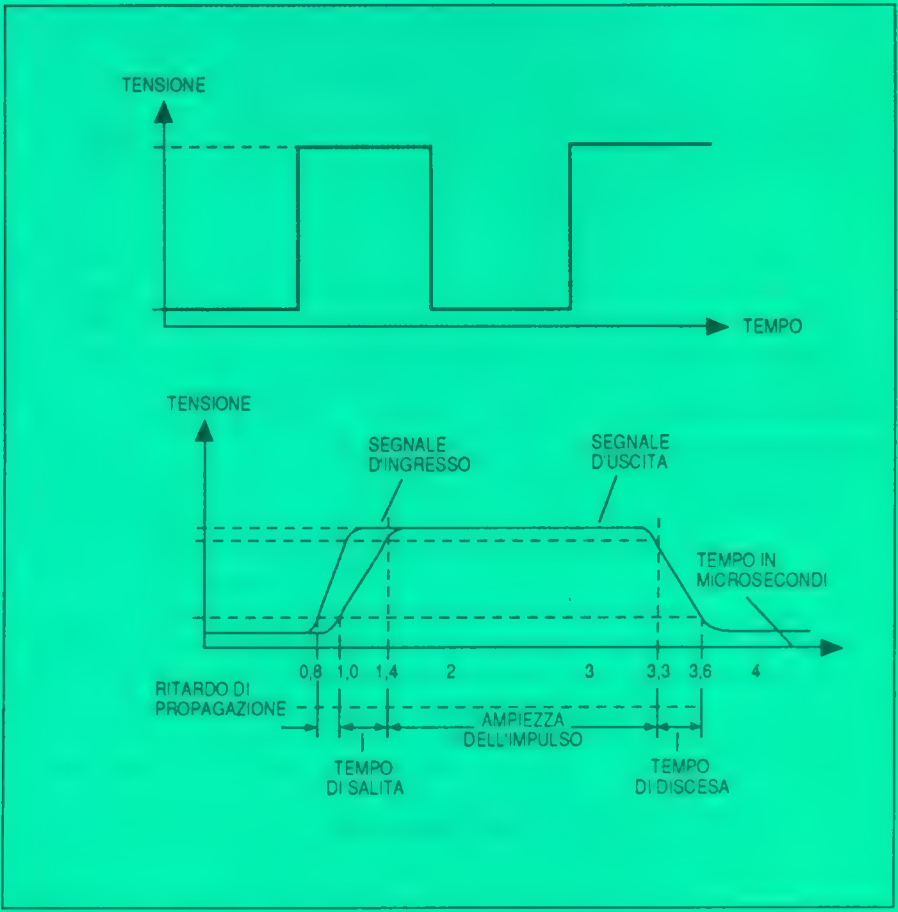

TENSIONE
I VALORI DELLA TENSIONE VARIANO CON CONTINUITA' IN FUNZIONE DEL TEMPO
+V
0
-V
TEMPO

Figura 1.4.
Segnale digitale


TENSIONE
TEMPO
TENSIONE
SEGNALE D'INGRESSO
SEGNALE D'USCITA
TEMPO IN MICROSECONDI
0,8
1,0
1,4
2
3
3,3
3,6
4
RITARDO DI PROPAGAZIONE
AMPIEZZA DELL'IMPULSO
TEMPO DI SALITA
TEMPO DI DISCESA

I segnali digitali rappresentano informazioni sotto forma di codice utilizzando i livelli di tensione per rappresentare i valori binari 0 o 1, in diverse combinazioni. La transizione tra i livelli di tensione viene ignorata mentre vengono considerati solo gli stati 0 o 1.

un segnale digitale varia nel tempo, ma il suo livello può assumere solo uno o due valori di tensione per rappresentare i numeri binari 0 e 1, come mostrato in *figura 1.4a*. Quando il segnale è al livello di tensione o corrente che rappresenta il valore 1, si ha il valore logico 1 e quando il segnale è al livello che rappresenta lo 0 si ha il valore logico 0. Naturalmente, ci sono delle transizioni (cambiamenti) tra i livelli 0 e 1, ma il sistema digitale deve essere progettato in modo tale da ignorare il livello di tensione durante i passaggi tra 0 e 1 e viceversa. I livelli di tensione che rappresentano lo stato logico 0 e 1 dipendono dal tipo di dispositivi usati nei circuiti digitali. Ciò verrà spiegato più avanti.

MISURAZIONE DI SEGNALI COMPLESSI

Solitamente le variazioni dei livelli di tensione nei segnali digitali o in quelli analogici non sinusoidali avvengono ad altissima velocità, così che risulta impossibile la determinazione delle caratteristiche del segnale con semplici voltmetri o amperometri. Per poter delineare e misurare tali segnali rapidamente variabili e tali forme d'onda complesse si devono usare strumenti di misura speciali. Uno di questi è l'oscilloscopio.

Oscilloscopio

L'oscilloscopio è uno speciale strumento di prova per visualizzare segnali digitali ed analogici. Le variazioni in funzione del tempo di questi segnali elettrici vengono visualizzate su un display CRT, permettendo di misurare parametri del segnale come l'ampiezza, il periodo e la fase.

L'oscilloscopio visualizza il grafico dell'ampiezza del segnale in funzione del tempo contemporaneamente al segnale stesso, anche se il segnale dura solo alcuni milionesimi o miliardesimi di secondo. L'oscilloscopio viene inserito nel circuito esattamente come un voltmetro e può essere usato per misure di ampiezza, di periodo e di fase. Le misure che interessano vengono lette dai grafici che appaiono sullo schermo del tubo a raggi catodici (o, più brevemente, CRT, da cathode-ray tube) dell'oscilloscopio, piuttosto che dalla posizione dell'ago di uno strumento analogico o dalle cifre che appaiono sul display di un voltmetro digitale. Una quadrettatura, detta griglia, riprodotta sullo schermo, unitamente a controlli graduati, permette di effettuare misure accurate. L'oscilloscopio funziona allo stesso modo di un normale televisore, solamente visualizza sul display l'andamento di segnali elettrici invece che un'immagine.

Regolando i comandi dell'oscilloscopio si possono misurare particolari molto minuti dei segnali elettrici. Per esempio, scegliendo una scala dei tempi abbastanza piccola il primo impulso mostrato in *figura 1.4a* assomiglierà a quello in *figura 1.4b*. Infatti tutti i segnali digitali hanno le caratteristiche mostrate in *figura 1.4b* quando la scala dei tempi è stata ridotta abbastanza da poter determinare il tempo necessario per passare dal livello 0 al livello 1 e viceversa. Una misurazione importante per un circuito digitale è quanto velocemente esso risponde ad un segnale di ingresso che lo faccia passare dallo stato logico 0 a quello 1 o viceversa. Questi tempi sono indicati in *figura 1.4b* come tempi di salita e di discesa rispettivamente. Misure dei tempi di salita e di discesa si ottengono facilmente quando l'onda digitale è rappresentata come mostrato in *figura 1.4b*.

Tempo di salita

Il tempo necessario per andare dal punto al 10% al punto al 90% di una transizione da 0 ad 1 di una forma d'onda viene detto tempo di salita.

Il tempo di salita è definito come il tempo che intercorre tra il punto al 10% del fronte di transizione e il punto al 90% del fronte stesso. Questi punti sull'onda possono essere localizzati con accuratezza mediante le linee di calibrazione delle ampiezze sulla griglia. Quindi il tempo tra questi due punti può essere letto sulla scala dei tempi riportati sulla griglia. In *figura 1.4b* il punto al 90% cade a 1,4 milionesimi di secondo o 1,4 microsecondi e il punto al 10% si ha a 1 microsecondo; di conseguenza il tempo di salita è 1,4 -1,0, cioè 0,4 microsecondi. Perciò il segnale passa dal livello 0 al livello 1 in 0,4 microsecondi. Un altro modo di esprimere questo numero è 400 miliardesimi di secondo o 400 nanosecondi.

Tempo di discesa

Il tempo di discesa di una forma d'onda è il tempo tra il punto al 90% e il punto al 10% di una transizione da 1 a 0.

Il tempo di discesa è il tempo impiegato dal segnale per scendere dal punto al 90% a quello al 10% nella transizione da 0 a 1. Anch'esso può essere determinato facilmente dalle informazioni contenute in *figura 1.4b*. Il punto al 90% cade in corrispondenza di 3,3 microsecondi e il punto al 10% cade a 3,6 microsecondi sulla scala dei tempi. Perciò il tempo di discesa è la differenza tra questi due tempi, e risulta essere di 0,3 microsecondi o 300 nanosecondi. Quando si devono misurare questi ed altri parametri di segnali impulsivi, l'impulso in ingresso deve avere parametri migliori di quelli del dispositivo o del circuito che è esaminato. Per esempio, se il dispositivo testato ha un tempo di salita di 400 nanosecondi, allora il tempo di salita dell'impulso di ingresso deve essere meno di 400 nanosecondi.

Ritardo di propagazione

I circuiti digitali hanno un ritardo di propagazione - il tempo che impiega il segnale per propagarsi dall'ingresso all'uscita. Tali ritardi si accumulano col passaggio del segnale da un circuito ad un altro.

Un altro tempo importante nei circuiti digitali è il ritardo di propagazione. Si definisce così il tempo che intercorre tra l'istante in cui il segnale d'ingresso inizia a cambiare e quello in cui inizia a cambiare il segnale in uscita. Tutti i dispositivi in un circuito, come pure i collegamenti tra di essi, introducono un ritardo di propagazione. Poiché il ritardo di ogni componente si somma ai ritardi di tutti gli altri in quanto il segnale passa attraverso diversi dispositivi, il ritardo totale può essere significativo in circuiti che abbiano una sincronizzazione critica. Questo punto sarà discusso in dettaglio in un capitolo successivo.

Il ritardo di propagazione si misura dal punto al 10% dell'impulso in ingresso al punto al 10% di quello in uscita come mostrato in *figura 1.4b*. In questo esempio, il ritardo di propagazione è 1,0 - 0,8 = 0,2 microsecondi o 200 nanosecondi.

Caratteristica di trasferimento

Il tempo di transizione (o commutazione) tra i livelli logici e i valori delle tensioni di soglia per i livelli logici 0 ed 1 sono molto importanti. La transizione può essere rappresentata come una funzione del tempo per mostrare quanto è veloce o lento un circuito nell'eseguire questo cambiamento di stato. La commutazione può essere descritta da un grafico come quello in *figura 1.5* per un dato tipo di circuito, in cui la

La transizione tra i livelli logici 0 ed 1 determina la velocità di funzionamento del circuito.

tensione di uscita è espressa in funzione della tensione di ingresso. Questo grafico è detto caratteristica di trasferimento del dispositivo.

Idealmente, per operazioni digitali, la curva della tensione dovrebbe presentare un passaggio rapido da un livello all'altro, ad un certo punto, detto soglia logica, posto a metà tra i livelli logici 0 ed 1. In pratica, la curva presenta una certa inclinazione nella zona di transizione. In generale un livello di ingresso o di uscita si considera come 0 logico se è al di sotto del livello dove inizia la zona inclinata della curva di transizione. In *figura 1.5* questo accade per qualunque tensione d'ingresso minore di 0,8 volt, mentre un valore tipico è 0,4 volt. Analogamente, si ha un valore 1 in ingresso quando la tensione in ingresso sta sopra (cioè alla destra) alla zona inclinata della curva di transizione. In *figura 1.5* si ha un 1 al di sopra di 2,0 volt, con un valore tipico di 2,4 volt. Il punto medio della zona inclinata della regione di transizione viene detto soglia logica. Nell'esempio tale punto è a 1,4 volt

Famiglie di dispositivi

Ci sono diverse famiglie di dispositivi digitali disponibili e ciascuna di esse usa livelli di tensione differenti per indicare l'1 logico e lo 0 logico. Questi livelli logici possono essere tensioni positive o negative.

I valori citati precedentemente sono i tipici valori associati alla nota famiglia di dispositivi TTL (Transistor-Transistor-Logic). Comunque, i livelli di tensione corrispondenti ai livelli logici 0 e 1 variano considerevolmente da una famiglia all'altra e i livelli relativi a tre importanti famiglie logiche sono riportati in *figura 1.6*. I livelli TTL, ripresi dalla *figura1.5*, mostrano che entrambi i livelli logici sono rappresentati da tensioni positive. Al contrario, la famiglia ECL (Emitter Coupled Logic) ha una soglia di 0 volt. Il valore 1 è rappresentato da qualunque tensione superiore a0,2 volt con il livello nominale di 0,4 volt. Uno 0 è rappresentato da qualunque tensio-

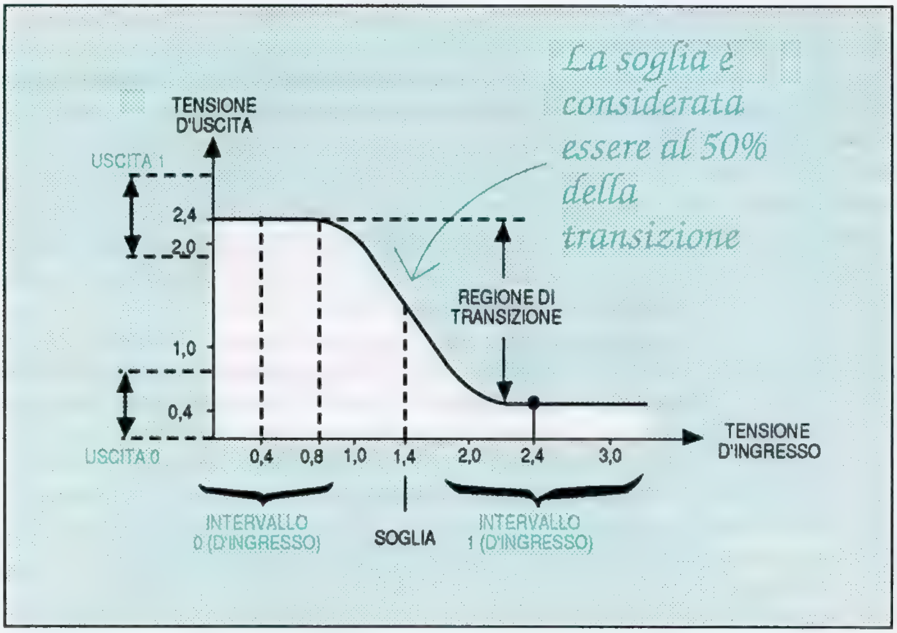


Figura 1.5. Caratteristiche tipiche di trasferimento

Figura 1.6. Livelli di segnale per tre famiglie logiche

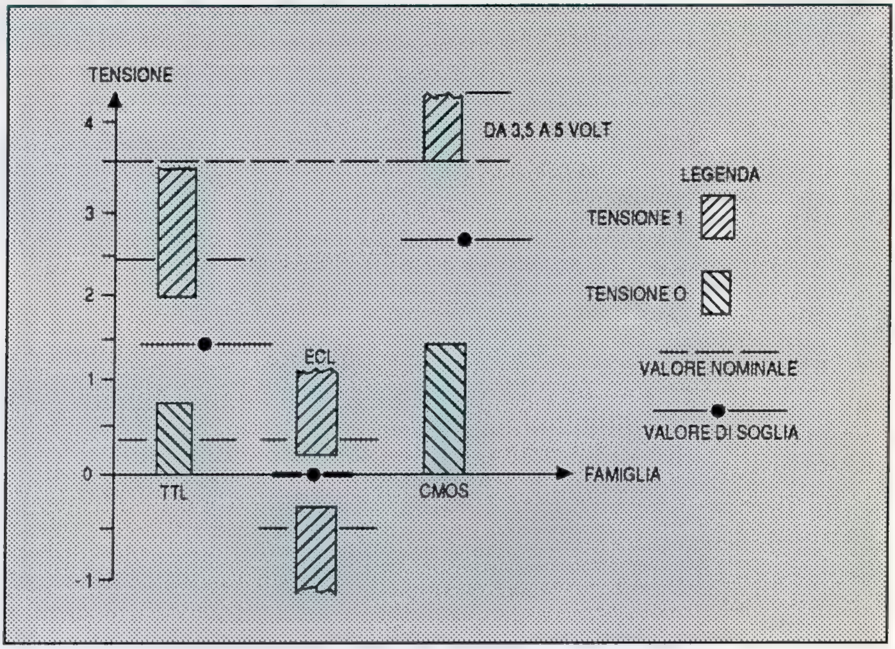
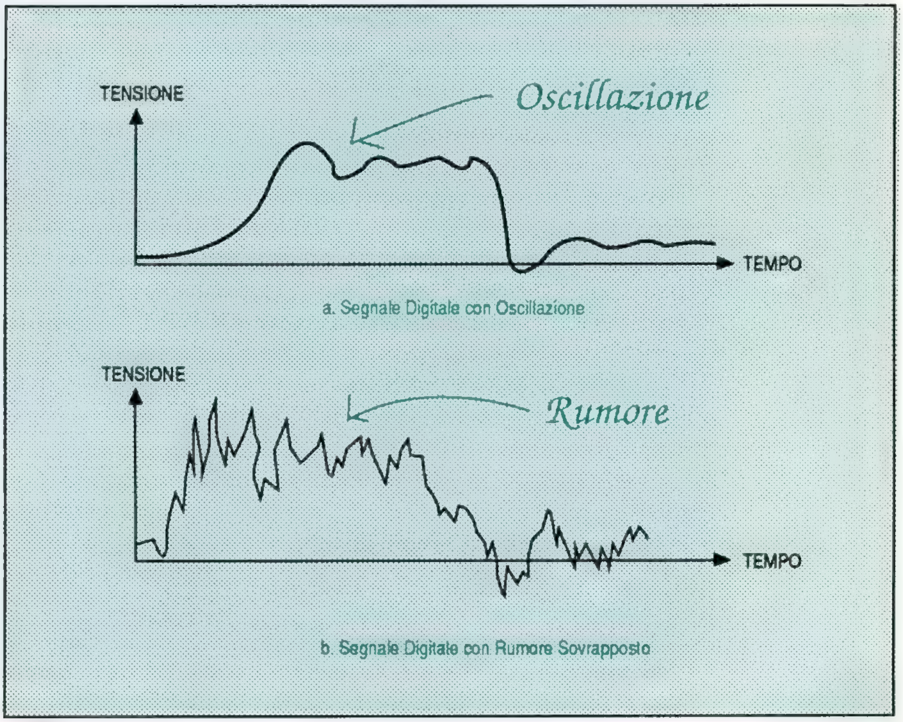


Figura 1.7. Segnali digitali indesiderati



a. Segnale Digitale con Oscillazione

b. Segnale Digitale con Rumore Sovrapposto ne negativa maggiore, in modulo, di –0,2 volt, con un livello nominale di –0,4 volt. La famiglia CMOS (Complementary Metal Oxide Semiconductor) è molto diversa dalla famiglia ECL. Gli invertitori e le porte CMOS hanno una caratteristica di trasferimento quasi ideale (squadrata con rapidissime variazioni di tensione nella zona di transizione) con i livelli logici che sono funzione della tensione di alimentazione utilizzata. In generale il valore della soglia è metà del valore della tensione di alimentazione. I valori mostrati in *figura 1.6* sono relativi ad un invertitore o ad una porta CMOS operante con un'alimentazione di 5 volt. La soglia è di 2,5 volt, il valore logico 1 si ha da 3,5 a 5 volt, il valore 0 da 0 a 1,5 volt.

Distorsione del segnale

I segnali digitali reali in un circuito funzionante hanno raramente la perfetta forma rettangolare mostrata in *figura 1.4a*. Un'onda più simile a quella che si ritrova nei circuiti reali è mostrata in *figura 1.7a*. Si noti che il segnale passa attraverso la zona di transizione, ma supera il livello 0 o 1 desiderato. Ritorna poi verso il livello desiderato, ma lo supera nuovamente in direzione opposta. Questa oscillazione intorno al valore voluto, chiamata oscillazione transitoria, continua finché il segnale non si attesta sul valore desiderato. L'oscillazione transitoria è causata dagli elementi accumulatori di energia (induttori e condensatori) nei circuiti. Infatti, anche i tempi di salita e di caduta mostrati in *figura 1.4b* sono prodotti da elementi che immagazzinano energia nei collegamenti dei circuiti o nei componenti elettrici stessi.

Elementi che accumulano energia

La capacità in un dispositivo o in un circuito immagazzina l'energia del campo elettrostatico. Questa energia si oppone alle variazioni di tensione e tenta di mantenere il livello della tensione costante, cioè pari al valore immagazzinato. La capacità è la grandezza che provoca la graduale diminuzione o il graduale aumento del livello di tensione in ogni nodo o punto di misurazione all'interno del circuito. Un apposito componente, chiamato condensatore, è usato talvolta nei circuiti con lo scopo di immagazzinare energia; comunque, anche se non ci sono condensatori collegati al circuito, esiste ugualmente una certa capacità parassita che immagazzina energia, associata ai fili di collegamento o presente fra elementi di dispositivi come i transistori. I fili di collegamento possiedono anche un'induttanza che immagazzina l'energia del campo magnetico. Quest'energia accumulata si oppone alle variazioni di corrente e cerca di mantenere la corrente nel filo ad un livello costante. Così l'energia immagazzinata in un campo magnetico provoca il passaggio della corrente ancora per un breve tempo dopo che la corrente di controllo ha terminato di fluire.

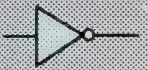
Rumore

Il rumore elettrico che si sovrappone ai livelli digitali può portare un

L'onda in *figura 1.7b* mostra un rumore (tensione che ha un'ampiezza casuale) sommato al segnale digitale desiderato. In un circuito che contenga transistori capaci di commutare molto velocemente, ognuna delle ampie oscillazioni può attraversare la soglia logica ed essere scambiata per un valore isolato di 0 o 1. Perciò il sistema po-

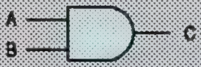
Figura 1.8. Operazioni digitali fondamentali

INGRESSI	USCITA
0	1
1	0



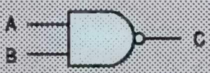
a. Tabella della Verità e Simbolo di un Invertitore

INGRESSI		USCITA
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1



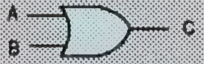
b. Tabella della Verità e Simbolo di una Porta AND

INGRESSI		USCITA
A	B	C
0	0	1
0	1	1
1	0	1
1	1	0



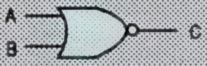
c. Tabella della Verità e Simbolo di una Porta NAND

INGRESSI		USCITA
A	B	C
0	0	0
0	1	1
1	0	1
1	1	1



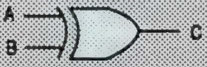
d. Tabella della Verità e Simbolo di una Porta OR

INGRESSI		USCITA
A	B	C
0	0	1
0	1	0
1	0	0
1	1	0



e. Tabella della Verità e Simbolo di una Porta NOR

INGRESSI		USCITA
A	B	C
0	0	0
0	1	1
1	0	1
1	1	0



f. Tabella della Verità e Simbolo di una Porta OR Esclusivo (XOR) sistema a rispondere in maniera errata, alterando il funzionamento del sistema.

trebbe ritenere che il segnale è cambiato diverse volte invece che una sola, la qual cosa, naturalmente, sconvolgerebbe il funzionamento del sistema stesso. Per assicurare che il funzionamento del sistema sia affidabile, i circuiti devono essere progettati in modo che non rispondano ad un rumore sovrapposto al segnale, oppure in modo che il rumore sia ridotto, o meglio ancora in modo da soddisfare entrambe queste condizioni. Su questi fattori verrà posta particolare attenzione nei capitoli successivi di questo libro.

Il modo in cui un circuito risponde ad un segnale di oscillazione transitoria o ad un rumore dipende in larga misura dal tipo di dispositivi usati. Come già detto, ci sono diversi tipi di circuiti digitali o famiglie di componenti logici usati nei sistemi digitali attuali. Le differenze principali tra i vari tipi di dispositivi sono legate ai componenti di cui fanno uso e ai livelli della tensione di funzionamento. Questi fattori influenzano il tempo di salita e di discesa, il ritardo di propagazione, la gamma di tensione per i valori logici 0 e 1, la sensibilità del dispositivo al rumore e le correnti che entrano ed escono da esso nelle condizioni di 0 e 1. Queste proprietà e le principali caratteristiche delle più comuni famiglie di componenti logici saranno esaminate in dettaglio nel capitolo 3. Quale che sia la famiglia usata in un sistema digitale, certe operazioni digitali fondamentali sono disponibili in tutte le famiglie. Queste operazioni saranno trattate in seguito.

OPERAZIONI DIGITALI FONDAMENTALI

La maggior parte delle operazioni logiche digitali sono ottenute dalla combinazione di quattro funzioni logiche fondamentali: invertitore, AND, OR, e OR Esclusivo.

Invertitori

Il livello logico d'uscita di un invertitore è sempre opposto al livello logico d'ingresso. Se l'ingresso è 1, l'uscita è 0 e viceversa.

Il simbolo dell'invertitore usato negli schemi o nei diagrammi logici è mostrato in *figura 1.8a*. Il piccolo cerchio sull'uscita indica inversione. (A volte il cerchietto è sull'ingresso piuttosto che sull'uscita.) Il funzionamento in termini di valori logici 0 o 1 è mostrato anch'esso nella tabella della verità in *figura 1.8a*. La tabella della verità si utilizza per riassumere le funzioni logiche delle porte e degli elementi digitali. Essa descrive lo stato dell'uscita per ogni combinazione degli stati d'ingresso. Come si vede da questa tabella l'operazione dell'invertitore è quella che il suo nome implica, cioè l'uscita è l'inversione logica o l'opposto dell'ingresso. Se l'ingresso è uno 0 logico, l'uscita è un 1 logico.

Porta AND

L'uscita di una porta AND sarà un 1 logico solo se entrambi gli ingressi sono un 1 logico: in ogni altro caso si avrà in uscita uno 0 logico.

La tabella della verità e il simbolo per la porta AND sono mostrati in *figura 1.8b*. Si può riassumere il funzionamento generale della porta dicendo che un'uscita 1 si avrà solo se tutti gli ingressi saranno al livello 1. Se un qualsiasi ingresso fosse invece 0 l'uscita sarebbe uno 0. Di conseguenza si usa questa porta per generare un'uscita soltanto nel caso in cui si verifichino tutte le condizioni d'ingresso richieste. La figura 1.8c mostra una porta derivata dalla NAND. Questa è la porta NAND che funziona come una porta AND con un invertitore in uscita. Così, se un qualsiasi ingresso di una

Una porta AND con un invertitore all'uscita è una porta NAND. Per un qualsiasi ingresso al livello 0 si avrà un 1 in uscita.

porta NAND è al livello 0, l'uscita sarà al livello 1. Soltanto nel caso in cui tutti gli ingressi siano al livello 1 si avrà un'uscita al livello 0. La porta NAND può essere usata nello stesso modo di una porta AND quando si desideri una condizione d'uscita che sia uno 0 invece che un 1.

Porta OR

In una porta OR se c'è un 1 in ingresso si avrà un 1 in uscita. La porta NOR relativa avrà un 1 in uscita solo quando tutti gli ingressi sono 0.

Un'altra operazione digitale fondamentale è quella della porta OR, mostrata nella *figura 1.8d*, e della sua funzione derivata, la porta NOR, mostrata in *figura 1.8e*. La porta OR ha un'uscita 1 se uno qualsiasi degli ingressi è 1. Avrà invece un'uscita 0 solo nel caso in cui tutti gli ingressi siano 0. La NOR, che può essere considerata una OR seguita da un invertitore, opera esattamente al contrario. Se uno qualsiasi degli ingressi è un 1, l'uscita della porta NOR sarà uno 0. Soltanto nel caso in cui tutti gli ingressi della porta NOR siano 0, l'uscita sarà 1.

Si noti che la OR si comporta, con un 1 in ingresso, come la NAND con uno 0 in ingresso; cioè, se uno qualsiasi degli ingressi raggiunge la condizione desiderata, l'uscita della porta sarà un 1. Analogamente, la NOR si comporta con un 1 in ingresso come la AND con uno 0 in ingresso; vale a dire che se uno qualsiasi degli ingressi raggiunge la condizione desiderata, l'uscita sarà uno 0. Dato che le porte sono logicamente equivalenti, qualsiasi sistema digitale può venire costruito usando solo porte NAND o porte NOR.

Porta OR Esclusivo

Un dispositivo utilizzato in circuiti binari addizionatori, la porta OR Esclusivo (XOR), fornirà un 1 solo se gli ingressi sono opposti.

Una funzione digitale frequentemente usata è quella della porta OR esclusivo (o, più brevemente, XOR da eXclusive-OR) mostrata nella *figura 1.8f*. In questo dispositivo si ha un 1 in uscita soltanto se gli ingressi sono diversi. Se invece i due ingressi sono uguali (entrambi 0 o entrambi 1), l'uscita sarà 0. La porta XOR è utile per confrontare un segnale digitale con un altro. È anche usata nei circuiti binari sommatori.

Tutti i dispositivi mostrati nella *figura 1.8* appartengono alla cosiddetta classe dei circuiti logici combinatori. Questi circuiti forniscono un'uscita che è una funzione dei valori d'ingresso presenti. In tali dispositivi non c'è memorizzazione di informazioni. Invece la memorizzazione è necessaria per molte operazioni digitali e quindi si sono sviluppati diversi metodi per supplire a questa funzione. La funzione del principale elemento di memorizzazione ad un bit sarà trattata qui di seguito.

MEMORIZZAZIONE DI INFORMAZIONI DIGITALI

Flip-Flop asincrono

Le porte combinatorie possono essere collegate fra loro in modo tale da formare un flip-flop R -

Dispositivi combinatori possono essere collegati fra loro in modo tale da fornire la memorizzazione di informazioni digitali come mostrato nella *figura 1.9*. La connessione principale è mostrata in *figura 1.9a*. (R sta per reset e S sta per set. Il trattino sul segnale si legge come "not"; per cui $\overline{R}$ va letto come "R not"). In questo circuito, supponendo che entrambi gli ingressi $\overline{S}$ ed $\overline{R}$ siano al livello 1, le porte agiscono semplicemente come invertitori. Così, se l'uscita Q è al livello 0 l'uscita $\overline{Q}$ è costret-

S, un fondamentale circuito digitale per la memorizzazione.

ta al livello 1. Il trattino sul nome del segnale significa che il suo livello logico è l'opposto (inversione o complemento) del nome del segnale senza il trattino. Questo assegna degli 1 ad entrambi gli ingressi della porta di destra, il che mantiene l'uscita Q al livello 0. Perciò il circuito memorizza uno 0 al terminale Q e il suo complemento, un 1, al terminale $\overline{Q}$.

In questo e in tutti i circuiti, se un simbolo di ingresso è segnato con un trattino il segnale è inattivo quando l'ingresso è ad 1 e attivo quando l'ingresso è a 0. Nella *figura 1.9a*, se si mette uno 0 all'ingresso $\overline{S}$, l'uscita Q è costretta ad 1 e l'uscita $\overline{Q}$ a 0. Le uscite resteranno in questo stato anche dopo che l'ingresso $\overline{S}$ sarà ritornato ad 1; di conseguenza l'uscita viene memorizzata. Ci si riferisce all'ingresso $\overline{S}$ come ad un ingresso attivo basso poiché un valore basso (lo 0) a questo ingresso fa sì che il circuito risponda all'ingresso. In questo caso il risultato è la memorizzazione di un 1 nel circuito al terminale Q (e di uno 0 al terminale $\overline{Q}$). Quindi l'ingresso $\overline{S}$ è un ingresso set-not che attiverà il circuito per memorizzare un 1 quando è portato al valore logico 0. Similmente, portare un ingresso $\overline{R}$ al valore basso ripristina il circuito così che Q è portato a 0 e memorizza uno 0 al suo terminale (mentre $\overline{Q}$ memorizza un 1). Questo circuito effettua l'operazione principale di memorizzare una sola cifra binaria (o, più brevemente, Bit da binary digit).

Un flip-flop $\overline{R}$ - S risponde ad ingressi attivi bassi (0), mentre un flip-flop R - S risponde ad ingressi attivi alti (1).

Il circuito è chiamato flip-flop per il modo in cui le uscite oscillano per mantenersi complementari. Il particolare circuito mostrato nella *figura 1.9a* viene chiamato flip-flop set-not / reset-not (flip-flop R-S) perché gli ingressi sono attivi con un ingresso 0. La *figura 1.9b* mostra il simbolo per il circuito e la *figura 1.9c* la sua tabella della verità. Si noti che $\overline{R}$ e $\overline{S}$ non possono essere entrambi a 0 contemporaneamente, perché ciò significherebbe comandare al flip-flop di memorizzare contemporaneamente uno 0 ed un 1, cosa che non può fare. Quando entrambi gli ingressi $\overline{S}$ ed $\overline{R}$ hanno valore 1, le uscite del flip-flop non cambiano; esse mantengono semplicemente la loro memorizzazione Q e $\overline{Q}$. Un valore basso su $\overline{S}$ fa memorizzare al circuito un 1, mentre un valore basso su R causa l'azzeramento del circuito e fa memorizzare al circuito stesso uno 0 sul terminale d'uscita Q.

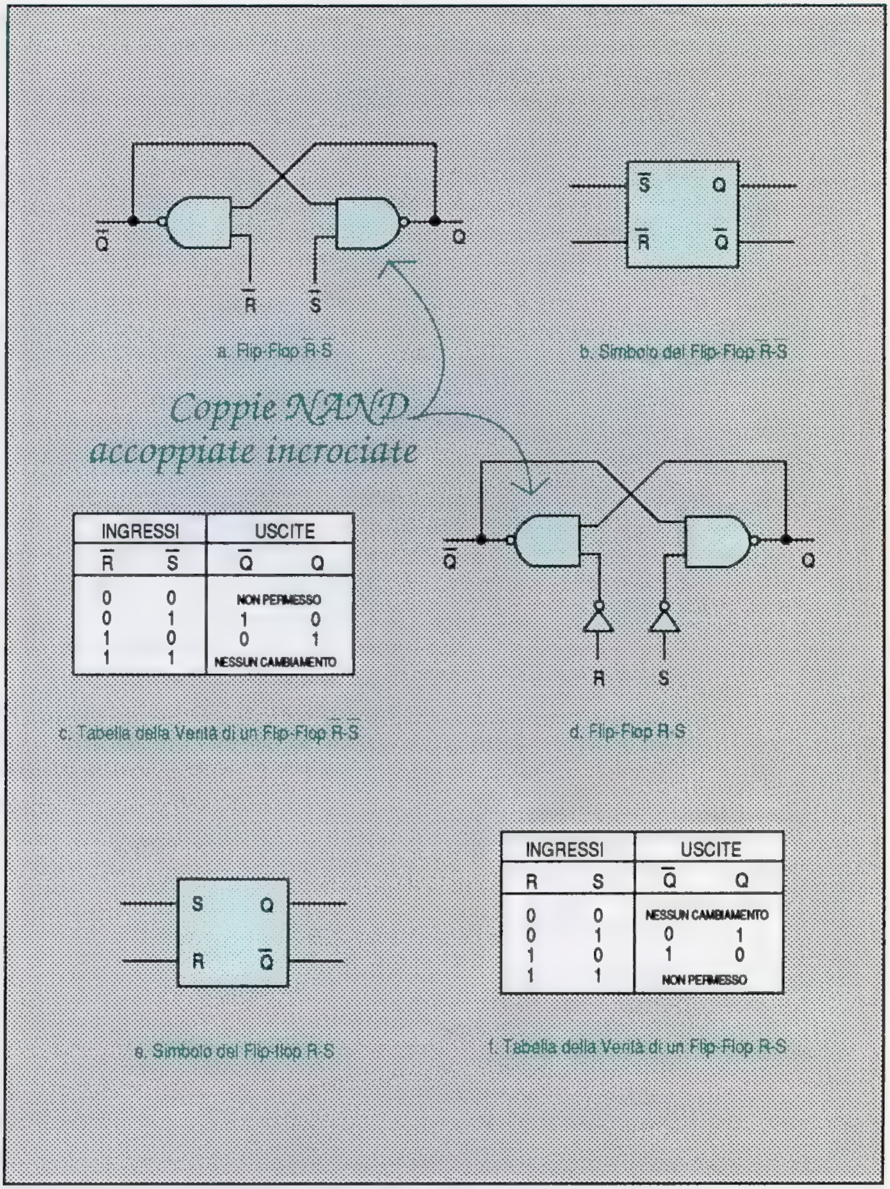
Se è necessario un flip-flop che risponda a segnali attivi alti (un 1 sulle linee d'ingresso) invece che a segnali attivi bassi, è possibile modificare a questo scopo il circuito della *figura 1.9a* aggiungendo degli invertitori agli ingressi come mostrato nella *figura 1.9d*. La *figura 1.9e* mostra il simbolo logico del flip-flop mentre la *figura 1.9f* mostra la sua tabella della verità. Questo flip-flop è un flip-flop set/reset (flip-flop R-S) che mantiene la memorizzazione fintanto che gli ingressi sono al livello 0. Un 1 all'ingresso S farebbe memorizzare al flip-flop un 1 sul terminale Q mentre un 1 sull'ingresso R gli farebbe memorizzare uno 0 sul terminale Q. Anche in questo caso, sia R che S non possono essere attivi (ad un livello 1) contemporaneamente.

Flip-flop sincrono

Un segnale di clock controlla quando un flip-flop sincrono deve rispondere ai segnali d'ingresso.

Il tipo di memorizzazione mostrato in *figura 1.9* è noto come memorizzazione asincrona, poiché l'uscita del flip-flop cambia immediatamente (a parte il ritardo di propagazione) quando viene applicato un segnale attivo su uno degli ingressi R o S. Un altro tipo di flip-flop fornisce un terminale di controllo per determinare quando il dispositivo deve memorizzare le informazioni in ingresso. Il terminale di controllo vie-

Figura 1.9. Semplici dispositivi di memoria asincroni


$\overline{Q}$
Q
$\overline{R}$
$\overline{S}$
$\overline{S}$
Q
$\overline{R}$
$\overline{Q}$
Coppie NAND accoppiate incrociate
$\overline{Q}$
Q
R
S
S
Q
R
$\overline{Q}$

a. Flip-Flop $\overline{R}$-$\overline{S}$

b. Simbolo del Flip-Flop $\overline{R}$-$\overline{S}$

INGRESSI		USCITE	
$\overline{R}$	$\overline{S}$	$\overline{Q}$	Q
0	0	NON PERMESSO	
0	1	1	0
1	0	0	1
1	1	NESSUN CAMBIAMENTO	

c. Tabella della Verità di un Flip-Flop $\overline{R}$-$\overline{S}$

d. Flip-Flop R-S

e. Simbolo del Flip-flop R-S

INGRESSI		USCITE	
R	S	$\overline{Q}$	Q
0	0	NESSUN CAMBIAMENTO	
0	1	0	1
1	0	1	0
1	1	NON PERMESSO	

f. Tabella della Verità di un Flip-Flop R-S ne chiamato strobe, clock, o latch ed il flip-flop viene chiamato flip-flop sincrono. La *figura 1.10* ne illustra il simbolo per un tipo di flip-flop, noto come flip-flop D. Il terminale d'ingresso è indicato da una D ed il clock da un triangolo. La relazione tra il segnale d'ingresso ed il segnale di uscita è mostrata nella *figura 1.10b.* È possibile che l'ingresso sul terminale D cambi, ma l'uscita Q non cambia finché il segnale di clock non passa dal livello 0 al livello 1. A questo punto l'uscita Q assume il valore che D aveva quando è avvenuta la transizione del segnale di clock. La transizione del segnale di clock da 1 a 0 o ulteriori cambiamenti dell'ingresso D non hanno altri effetti sul dato memorizzato all'uscita Q del flip-flop.

Un flip-flop attivato dal fronte di salita immagazzina dati quando il segnale di clock compie una transizione dal livello 0 al livello 1.

Poiché un dato è memorizzato quando l'ingresso di clock ha transizione da un livello basso a uno alto, questo viene chiamato flip-flop attivato dal fronte di salita (leading edge flip-flop oppure rising edge-triggered flip-flop). Esistono anche flip-flop che memorizzano quando il clock ha una transizione da un livello alto ad un livello basso: essi vengono chiamati flip-flop attivati dal fronte di discesa (trailing edge flip-flop oppure falling edge-triggered flip-flop). Questi hanno simboli logici uguali ad eccezione di un circoletto sul terminale del clock.

Le relazioni temporali tra il clock e i segnali di ingresso e di uscita (*figura 1.10b*) includono i parametri temporali del circuito, cioè: il tempo di salita, il tempo di discesa, il ritardo di propagazione, il tempo di set-up e il tempo di mantenimento. I tempi di salita, di discesa e i ritardi di propagazione sono stati definiti in *figura 1.4b.* I tempi di set-up e di mantenimento sono in relazione col tempo in cui i dati devono rimanere stabili prima e dopo la transizione del clock.

La *figura 1.11* mostra la tempificazione per un flip-flop R-S asincrono. Ci sono ancora considerazioni riguardanti il tempo di salita e di discesa e il ritardo di propagazione, ma poiché non c'è segnale di clock non ci sono né il tempo di set-up né il tempo di mantenimento. L'uscita risponde semplicemente agli ingressi S ed R immediatamente (a parte il ritardo di propagazione) non appena avviene un cambiamento.

DATI DIGITALI SERIALI E PARALLELI

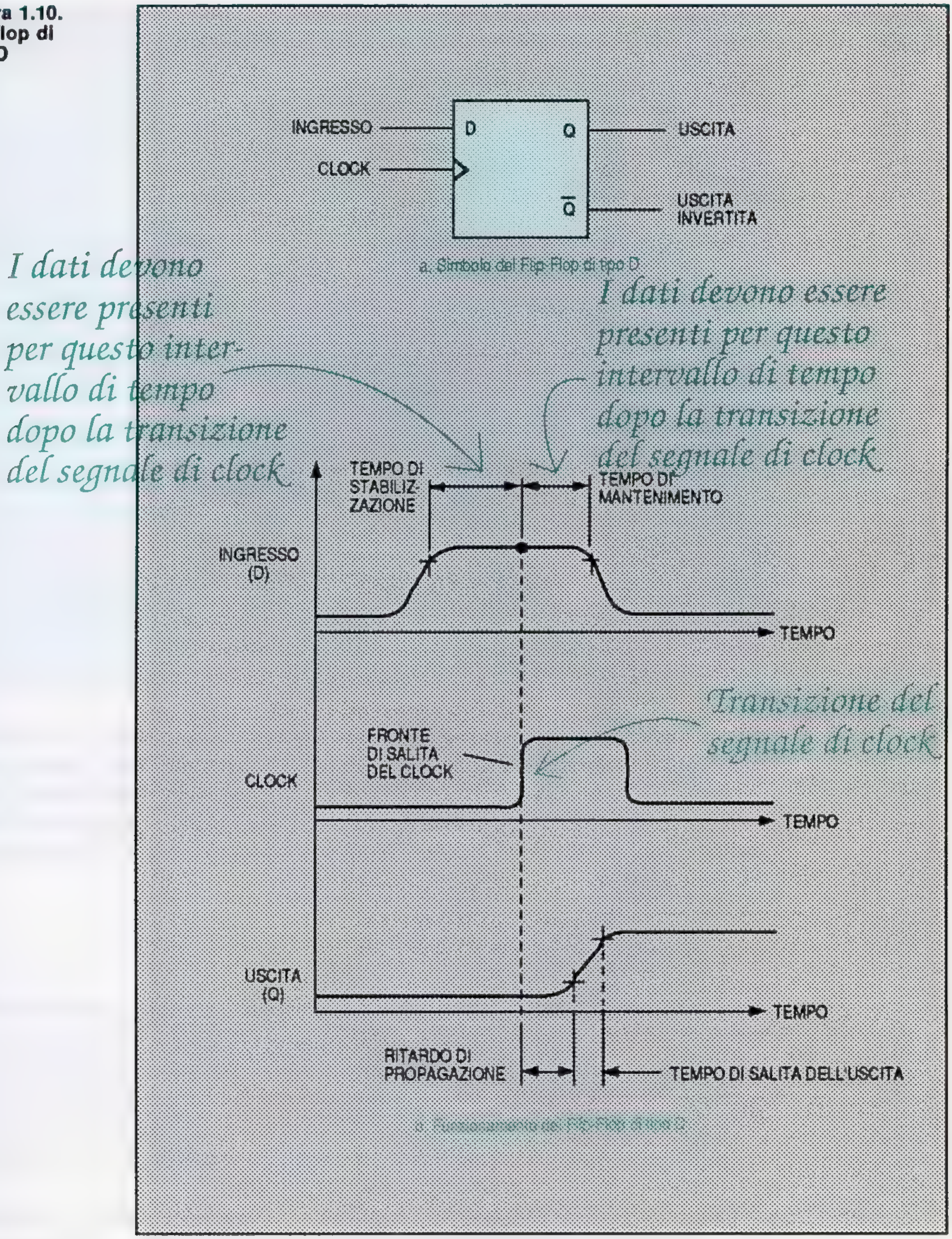
I dati digitali che percorrono da punto a punto un unico filo in un certo periodo di tempo vengono chiamati dati seriali. Se un certo insieme di 1 e di 0 rappresenta qualche informazione del sistema e deve essere trasmesso in forma seriale da una parte del sistema all'altra, il trasferimento richiede solo un'unica linea ed un'unica serie di circuiti trasmittenti e riceventi, come mostrato in *figura 1.12a.* Nella trasmissione seriale il bit meno significativo (il più a destra), d_0 , viene trasmesso per primo e quello più significativo (il più a sinistra), d_7 , viene trasmesso per ultimo.

La trasmissione seriale dei dati invia tutti i bit dei dati su di un singolo filo, in maniera sequenziale. È poco costosa, ma lenta.

La trasmissione di dati seriali è il metodo più economico per trattare le informazioni digitali, ma è anche il metodo più lento poiché ogni 0 ed ogni 1 nella sequenza richiede un certo tempo. Il tempo complessivo per trasmettere l'informazione dipende dal numero di bit e dal periodo di ciascun bit. Se per ogni bit occorrono 10 nanosecondi e vengono trasmessi 8 bit, per la trasmissione seriale occorreranno 80 nanosecondi.

Se questi stessi 8 bit vengono trasmessi in parallelo (tutti e 8 contemporaneamente) è necessario solo il tempo di un bit, 10 nanosecondi. Nella trasmissione dei dati in pa-

Figura 1.10. Flip-flop di tipo D



a. Simbolo del Flip-Flop di tipo D

b. Funzionamento del Flip-Flop di tipo D

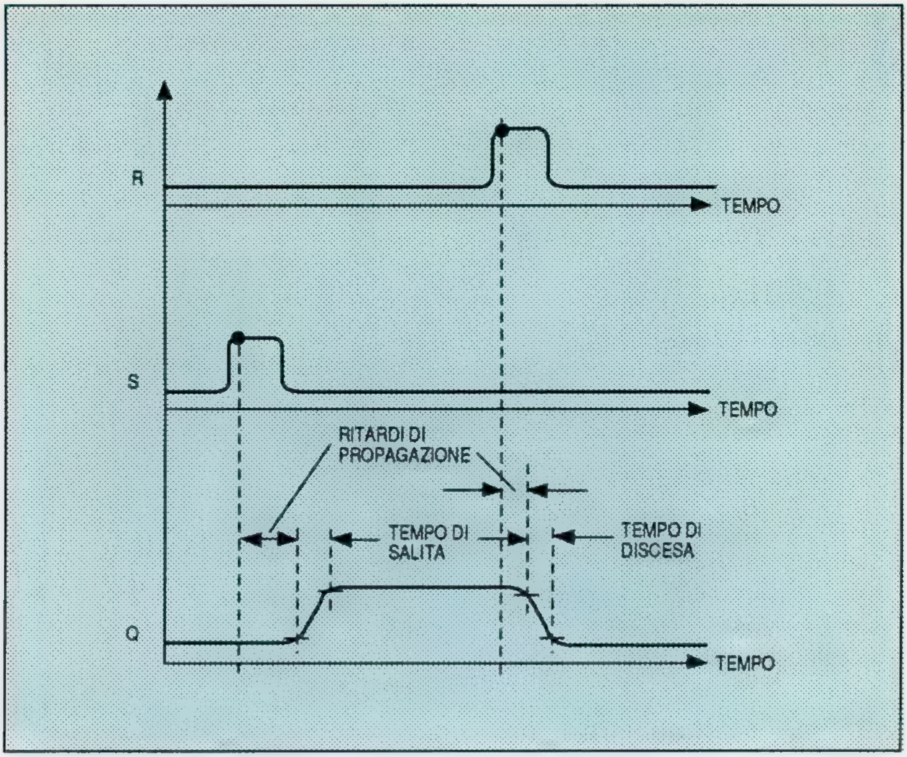
I dati devono essere presenti per questo intervallo di tempo dopo la transizione del segnale di clock

I dati devono essere presenti per questo intervallo di tempo dopo la transizione del segnale di clock

Transizione del segnale di clock

INGRESSO
CLOCK
D
Q
Q̄
USCITA
USCITA INVERTITA
TEMPO DI STABILIZZAZIONE
TEMPO DI MANTENIMENTO
INGRESSO (D)
TEMPO
FRONTE DI SALITA DEL CLOCK
CLOCK
TEMPO
USCITA (Q)
TEMPO
RITARDO DI PROPAGAZIONE
TEMPO DI SALITA DELL'USCITA

Figura 1.11. Flip-flop asincroni



La trasmissione dei dati in parallelo utilizza un filo per ciascun bit, inviando tutti i bit contemporanea–mente. È più veloce, ma più costosa, della trasmissione seriale dei dati.

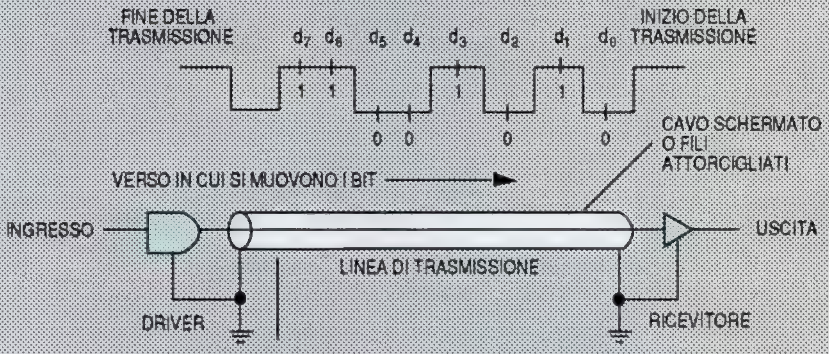
rallelo, per ogni bit vengono usati un circuito di trasmissione ed uno di ricezione su linee separate. Questo è mostrato nella *figura 1.12b*. Naturalmente, i circuiti di trasmissione in parallelo sono più costosi per la maggiore quantità di hardware necessaria. Il metodo scelto solitamente dipende o dal fatto che sia più importante la velocità operativa rispetto al costo dell'hardware o dall'esigenza di compatibilità con un altro sistema.

CODICI DIGITALI

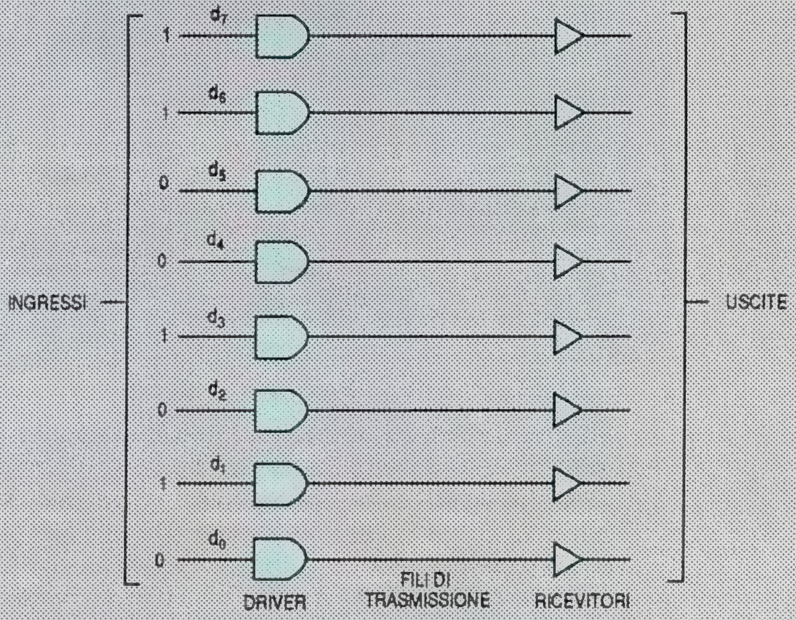
I codici digitali che utilizzano bit multipli con valori 1 o 0 possono rappresentare informazioni largamente diversificate.

Poiché i circuiti digitali usano solo i livelli di tensione 0 ed 1, per rappresentare un'informazione devono essere utilizzati codici composti da diverse combinazioni di bit. Che informazioni rappresenti un dato codice dipende dal suo uso all'interno del sistema digitale. Potrebbe rappresentare alcuni valori numerici, un comando o un'istruzione, un carattere alfabetico, o qualche sequenza di controllo on-off. Il significato del codice in queste applicazioni viene assegnato da alcune tavole di riferimento o dal modo in cui i circuiti del sistema sono connessi.
Il numero delle cose differenti che un codice può rappresentare dipende da quanti bit ci sono nel codice. Alcune possibilità sono illustrate nella *figura 1.13*. Se si usa solo un singolo bit, come mostrato in *figura 1.13a*, possono essere rappresentate solo due differenti azioni o numeri, come 0 ed 1, no e sì, oppure on e off. Utilizzando un codice a due bit possono essere usati quattro differenti numeri o comandi, come mo-

**Figura 1.12.
Trasmissione
digitale dei dati**


FINE DELLA TRASMISSIONE
d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0
INIZIO DELLA TRASMISSIONE
1 1 0 0 1 0 1 0
CAVO SCHERMATO O FILI ATTORCIGLIATI
VERSO IN CUI SI MUOVONO I BIT
INGRESSO
USCITA
LINEA DI TRASMISSIONE
DRIVER
RICEVITORE

a. Trasferimento Seriale di Dati del Codice Binario 11001010


d_7
d_6
d_5
d_4
d_3
d_2
d_1
d_0
1
1
0
0
1
0
1
0
INGRESSI
USCITE
DRIVER
FILI DI TRASMISSIONE
RICEVITORI

b. Trasferimento di Dati in Parallelo del Codice Binario 11001010

Figura 1.13. Semplici codici binari

Valori Dei Bit	Possibili Significati	
d_0	Numero	Istruzione
0	0	SPENTO
1	1	ACCESO

a. Codice a 1 Bit

Valori Dei Bit		Possibili Significati	
d_1	d_0	Numeri Decimali	Comandi
0	0	0	STOP
0	1	1	VAI AVANTI
1	0	2	VAI A DESTRA
1	1	3	VAI A SINISTRA

b. Codice a 2 Bit

Valori Dei Bit			Numeri Decimali	Istruzione
d_2	d_1	d_0		
0	0	0	0	SOMMA
0	0	1	1	SOTTRAI
0	1	0	2	MOLTIPLICA
0	1	1	3	DIVIDI
1	0	0	4	VALORE ASSOLUTO
1	0	1	5	NEGA
1	1	0	6	INCREMENTA
1	1	1	7	DECREMENTA

c. Codice a 3 Bit

strato in *figura 1.13b*, in quanto con due bit sono possibili quattro diverse loro combinazioni. Con un codice a tre bit, come mostrato in *figura 1.13c*, possono essere rappresentati otto numeri o istruzioni differenti. In ogni caso, il numero delle diverse combinazioni di 0 ed 1 disponibili per un codice ad n-bit è pari a 2^n . Perciò per un codice a due bit si hanno 2^2 , o 4 differenti combinazioni di 0 ed 1. Analogamente, per un codice a tre bit è 2^3 o 8 il numero di differenti combinazioni di 0 e 1. Un codice a 8 bit può rappresentare 256 combinazioni e un codice a 16 bit può rappresentare 65.536 combinazioni.

BCD

Il metodo BCD utilizza un codice a 4 bit per rappresentare le cifre decimali da

Un codice digitale comunemente usato è il codice a 4 bit di *figura 1.14*, che può rappresentare i numeri da 0 a 15 nel sistema numerico decimale (base 10) o i caratteri alfanumerici da 0 a 9 e da A ad F in un sistema numerico esadecimale (base 16). Se un codice a 4 bit è limitato alle configurazioni da 0000 a 1001, i numeri esadecima-

Figura 1.14. Cifre esadecimali (in base 16) e loro equivalenti binarie

Codice Binario	Cifra Esadecimale	Codice Binario	Cifra Esadecimale
0000	0	1000	8
0001	1	1001	9
0010	2	1010	A
0011	3	1011	B
0100	4	1100	C
0101	5	1101	D
0110	6	1110	E
0111	7	1111	F

Figura 1.15. Conversione di un numero da binario a decimale

Posizione Del Bit								
n	7	6	5	4	3	2	1	0
2^n	128	64	32	16	8	4	2	1
NUMERO BINARIO	0	1	0	1	0	0	1	1
VALORE DEL BIT x 2^n	0	64	0	16	0	0	2	1

SOMMA DEI PRODOTTI = 64 + 16 + 2 + 1 = 83
PERCIO': $01010011_2 = 83_{10}$
(L'INDICE INDICA LA BASE DEL NUMERO)

Figura 1.16. Conversione di un numero da binario ad esadecimale e da esadecimale a decimale

POSIZIONE CIFRA ESADECIMALE	1	0
NUMERO BINARIO	0101	0011
CIFRE ESADECIMALI EQUIVALENTI	5	3
16^n	16	1
VALORE CIFRA ESADECIMALE x 16^n	80	3

SOMMA DEI PRODOTTI = 80 + 3 = 83
PERCIO': $01010011_2 = 53_{16} = 83_{10}$

0 a 9. È semplice da usare, ma non è efficiente perché possono essere utilizzate solo 10 delle 16 possibili combinazioni a 4 bit.

li e decimali sono limitati ai numeri da 0 a 9. Perciò un modo semplice per rappresentare i numeri decimali è quello di utilizzare solo le prime dieci combinazioni di *figura 1.14* e sostituire ogni cifra nel numero con il suo codice a 4 bit. Quindi, ogni qualvolta venga ricevuto un codice a 4 bit, questo può essere decodificato invertendo il procedimento e sostituendo il codice a 4 bit con il suo corrispondente valore numerico decimale. Questo metodo di rappresentazione dei numeri decimali con gli equivalenti codici binari a 4 bit viene chiamato codice BCD (Binary Coded Decimal). Questo codice è semplice ma inefficiente, in quanto utilizza solo 10 delle possibili 16 combinazioni di un codice a 4 bit.

Numeri binari

I numeri rappresentati in un codice binario vengono utilizzati direttamente dai microprocessori o dagli altri sistemi digitali.

Per evitare questo ed altri problemi con il codice BCD molti sistemi digitali, particolarmente quelli che impiegano i microprocessori, utilizzano direttamente il codice binario per rappresentare i numeri. Il codice binario può rappresentare un numero binario (base 2) nello stesso modo in cui un codice decimale rappresenta numeri decimali (base 10). È un'operazione semplice convertire i numeri dalla base 10 alla base 2 e viceversa, in modo che sia possibile utilizzare pienamente la potenzialità del codice binario. Quando si sta lavorando con più di un sistema numerico per volta, la base di un particolare numero, se non è identificata a parole, viene indicata da un pedice immediatamente dopo l'ultima cifra: per esempio, 1001_2 oppure 9_{10}.

Conversione di un numero da binario a decimale

Nella conversione di un numero da binario a decimale, il valore di ciascuna cifra binaria viene moltiplicato per 2^N Il valore di N dipende dalla posizione della cifra nel codice. La somma di questi prodotti è il numero decimale.

La conversione di un numero binario in un numero decimale è illustrata in *figura 1.15* per il numero binario a 8 bit 01010011. Le posizioni delle cifre binarie sono identificate come bit da 0 a 7 partendo dall'ultimo bit a destra. Il bit 0 è il bit meno significativo e il bit 7 è quello più significativo. Ciascun valore della cifra binaria (0 o 1) viene moltiplicato per 2^n dove n è la posizione del bit. Questi prodotti vengono sommati e la loro somma è il numero decimale. Nell'esempio di *figura 1.15* solo i bit 0, 1, 4 e 6 hanno il valore 1; perciò 1 viene moltiplicato per la potenza di 2 per ognuna delle cifre (1, 2, 16 e 64) e questi prodotti vengono sommati per fornire il risultato, 83. Così, 83 è l'equivalente decimale del numero binario 01010011.

Conversione di un numero da binario a esadecimale

I numeri binari con più di 8 cifre utilizzano generalmente i numeri esadecimali come gradino intermedio nella conversione. La conversione è semplice.

Questo procedimento può essere utilizzato per ogni numero binario formato da qualsiasi numero di cifre, nella misura in cui le corrispondenti potenze di 2 sono conosciute o possono essere calcolate. Comunque, per numeri binari con più di 8 cifre è più conveniente utilizzare il procedimento che impiega i numeri esadecimali (base 16) come risultato intermedio. L'equivalente esadecimale di un numero binario viene determinato sostituendo ciascun gruppo di 4 cifre binarie, cominciando dagli ultimi 4 bit a destra, con la loro corrispondente cifra esadecimale. Per il numero binario 0000 0000 0101 0011, il numero in base 16 è 0053 che può anche essere scritto come 53_{16} Quindi si può utilizzare un procedimento simile a quello di *figura 1.15*, ma utilizzando potenze di 16 (invece di potenze di 2 in quanto ciascuna cifra è una cifra in base 16) come mostrato in *figura 1.16*. In questo caso la posizione 0 è occupata dalla cifra 3, la posizione 1 dalla cifra 5, tutte le altre cifre sono 0. Per la posizione 0 si ha: $16^0 = 1$ e 1 x 3 è uguale a 3. Analogamente, per la posizione 1 si ha: $16^1 = 16$ e 16 x 5 è uguale a 80. Allora 80 più il valore 3 precedentemente determinato forniscono l'equivalente decimale 83. Questo procedimento è molto più semplice che non lavorare con i singoli bit binari, particolarmente per numeri binari con più bit. Se i procedimenti di *figura 1.15* e *1.16* sono estesi per codici binari di 8 bit, i numeri binari che vanno da 0000 0000 a 1111 1111 rappresentano i numeri esadecimali che vanno da 00 a FF e che a loro volta rappresentano 256 numeri decimali, da 0 a 255. Quanto detto finora è valido nel caso di un codice di numeri binari senza segno (tutti i numeri sono positivi).

Numeri negativi

Il bit più a sinistra di un numero binario viene utilizzato come bit di segno: se è 1 indica un numero negativo, se è 0 un numero positivo.

È possibile rappresentare anche i numeri decimali negativi con un codice binario utilizzando l'ultimo bit di sinistra del numero binario come bit di segno. Se il bit di segno è 0, il numero è positivo; se il bit di segno è 1, il numero è negativo. Anche in questo caso con un codice a 8 bit possono essere codificati 256 numeri, ma ora i numeri vanno da -128 a +127. (0 è un numero positivo). Il procedimento più semplice per trasformare un tipo di codice di numeri con segno nel suo equivalente decimale, noto come codice del complemento a 2 del numero, è il seguente:

- Se il bit di segno è 0, si utilizza la procedura mostrata in *figura 1.15* o in *figura 1.16* e si appone un segno + davanti al numero decimale.
- Se il bit di segno è 1, si converte il codice binario a quello esadecimale utilizzando la *figura 1.16*, si sottrae il codice esadecimale da FF e quindi si aggiunge 1. Si converte questo codice esadecimale al suo equivalente decimale e si appone un segno -.

Il procedimento è illustrato per codici di numeri sia positivi che negativi in *figura 1.17*. Il codice binario 0111 0100 è equivalente a 74_{16} ed è un numero positivo. L'equivalente decimale di questo numero si trova come (7 x 16) + 4, cioè + 116.
La trasformazione del numero binario 11000010 è un po' più complicata in quanto l'ultima cifra 1 a sinistra indica che si tratta di un numero negativo. Dapprima il numero binario si converte nel suo equivalente esadecimale C2. Poi si sottrae C2 da FF per ottenere 3D e si aggiunge 1 a 3D per ottenere 3E. L'equivalente decimale del numero esadecimale 3E è (3 x 16) + 14 = 62. Perciò l'equivalente decimale del numero esadecimale C2 è -62 se C2 rappresenta un codice binario con segno (complemento a 2). Con questo codice, i numeri binari da 0000 0000 a 0111 1111 (numeri esadecimali da 00 a 7F) rappresentano rispettivamente i numeri decimali da 0 a +127. I numeri binari da 1000 0000 a 1111 1111 (numeri esadecimali da 80 a FF) rappresentano rispettivamente i numeri decimali da -128 a -1.

Conversione di un numero da decimale a esadecimale

La conversione di un numero da decimale a esadecimale viene eseguita in modo migliore seguendo una serie di procedure.

L'inverso del procedimento di *figura 1.16* può essere utilizzato per trasformare un numero decimale nel suo equivalente esadecimale e quindi nel suo equivalente binario. Il procedimento è illustrato in *figura 1.18*. Il numero decimale viene dapprima diviso per il più grande numero che sia una potenza di 16 e che dia quoziente di 1 o maggiore di 1. Questo quoziente è la cifra esadecimale più significativa (quella più a sinistra). Il resto viene diviso dalla più grande potenza di 16 e il quoziente è la successiva cifra esadecimale significativa. Quando si ritrova come quoziente 0, il precedente resto o quoziente è l'ultima cifra esadecimale significativa. Un procedimento simile può essere utilizzato per i numeri decimali negativi. La conversione può essere verificata servendosi del procedimento di *figura 1.16* per trasformare nuovamente il risultato esadecimale nel numero decimale.

Codice alfanumerico

Il codice ASCII

Naturalmente, i valori numerici non sono le uniche informazioni necessarie in un si-

Figura 1.17. Conversione del codice binario con segno (complemento)

DA BINARIO
A
ESADECIMALE
A
DECIMALE

$$\underbrace{0111}\qquad\underbrace{0100}$$

$$\begin{array}{r} 7 \\ \times\ 16 \\ \hline 112 \end{array} \quad + \quad \begin{array}{r} 4 \\ \times\ 1 \\ \hline 4 \end{array} = 116$$

a. Numero Binario Positivo

DA BINARIO
A
ESADECIMALE

$$\underbrace{1100}_{C}\qquad\underbrace{0010}_{2}$$

$$FF_{16} - C2_{16} + 1 = 3D_{16} + 1 = 3E_{16}$$

DA ESADECIMALE
(3E) A
DECIMALE

$$\begin{array}{r} 3 \\ \times\ 16 \\ \hline 48 \end{array} \quad + \quad \begin{array}{r} 14 \\ \times\ 1 \\ \hline 14 \end{array} = 62$$

METTERE IL SEGNO PER INDICARE UN NUMERO NEGATIVO: – 62

b. Numero Binario Negativo

Figura 1.18. Conversione di un numero da decimale ad esadecimale e da esadecimale a binario

NUMERO DECIMALE DATO = 1132

Resto	**Divisore**	**Quoziente**	
1132	256	4	Cifra esadecimale più significativa
108	16	6	Successiva cifra esadecimale significativa
12	1	12	Cifra esadecimale meno significativa ($12 = C_{16}$)

Perciò: $1132_{10} = 46C_{16} = 0100\ 0110\ 1100_2$

Prova di riconversione in decimale
$46C_{16} = (4 \times 256) + (6 \times 16) + (12 \times 1) = 1024 + 96 + 12 = 1132_{10}$

è un codice binario a 7 bit che permette di rappresentare in forma binaria 128 diversi caratteri alfanumerici e comandi di controllo.

stema digitale. Il sistema può avere la necessità di rappresentare caratteri numerici ed alfabetici contemporaneamente, o potrebbe aver bisogno di rappresentare alcuni comandi di controllo in un sistema. Per l'informazione alfanumerica, molti sistemi utilizzano l' American Standard Code for Information Interchange (ASCII, che si pronuncia "aschi"). Il codice ASCII è un codice binario a 7 bit che può rappresentare 128 caratteri e operazioni di controllo differenti. Questo codice è rappresentato in *figura 1.19* con il codice binario rappresentato da numeri esadecimali. Perciò il codice per i caratteri ASCII CODE è la sequenza dei numeri esadecimali 41 53 43 49 49 20 (per la spaziatura) 43 4F 44 45. Il codice ASCII per il numero 39 è rappresentato dalle cifre esadecimali 33 39.

CONVERSIONE DEI SEGNALI

I segnali analogici devono essere trasformati in segnali digitali per poter essere utilizzati da sistemi digitali.

È stato precedentemente affermato che i segnali analogici possono essere codificati in segnali numerici in modo tale da poter essere elaborati da circuiti digitali. Fondamentalmente, la codifica consiste nell'assegnare valori numerici per rappresentare i livelli di tensione del segnale. Questi valori numerici non sono necessariamente i reali valori di tensione. Il segnale analogico viene misurato (campionato) dal circuito di codifica a intervalli di tempo regolari e il numero binario equivalente del livello della tensione viene emesso sotto forma di segnale digitale. Il circuito che compie questa operazione viene chiamato convertitore analogico-digitale (A/D). Il suo circuito complementare, il convertitore digitale-analogico (D/A), viene utilizzato per riconvertire il codice digitale in un segnale analogico.

Il concetto della conversione A/D e D/A è basato su un principio chiamato teorema del campionamento di Nyquist:

- se il segnale analogico viene campionato uniformemente ad una frequenza che sia almeno il doppio della sua frequenza più alta, allora il segnale originale può essere ricostruito dai campionamenti eseguiti.

Il teorema può essere descritto dall'equazione:

$$f_S \geq 2BW$$

f_S è la frequenza di campionamento e BW è la banda passante del segnale (che definisce la frequenza più alta).

Campionamento di un segnale analogico

La *figura 1.20* mostra come un segnale analogico varia in funzione del tempo. Gli intervalli di tempo in cui l'onda viene campionata sono graficamente sovrapposti alla figura e, come si può osservare, l'ampiezza del segnale non cambia di molto nel breve intervallo di tempo tra due misurazioni. Si ha dunque che la misurazione nell'istante del campionamento è una rappresentazione fedele del segnale per un breve intervallo di tempo da entrambi i lati del punto di campionamento.

Il teorema di Nyquist afferma

Ovviamente, più elevato è il numero dei campioni nel tempo T più l'onda verrà riprodotta accuratamente, ma, come affermato dal teorema di Nyquist, se il segnale

Figura 1.19. Codice standard americano per lo scambio di informazioni (Codice ASCII)

DEC	HEX	CHR	DEC	HEX	CHR	DEC	HEX	CHR	DEC	HEX	CHR
0	0	NUL	32	20	SP	64	40	@	96	60	
1	1	SOH	33	21	!	65	41	A	97	61	a
2	2	STX	34	22	"	66	42	B	98	62	b
3	3	ETX	35	23	#	67	43	C	99	63	c
4	4	EOT	36	24	$	68	44	D	100	64	d
5	5	ENQ	37	25	%	69	45	E	101	65	e
6	6	ACK	38	26	&	70	46	F	102	66	f
7	7	BEL	39	27	'	71	47	G	103	67	g
8	8	BS	40	28	(	72	48	H	104	68	h
9	9	HT	41	29	)	73	49	I	105	69	i
10	A	LF	42	2A	*	74	4A	J	106	6A	j
11	B	VT	43	2B	+	75	4B	K	107	6B	k
12	C	FF	44	2C	,	76	4C	L	108	6C	l
13	D	CR	45	2D	-	77	4D	M	109	6D	m
14	E	SO	46	2E	.	78	4E	N	110	6E	n
15	F	S1	47	2F	/	79	4F	O	111	6F	o
16	10	DLE	48	30	0	80	50	P	112	70	p
17	11	DC1	49	31	1	81	51	Q	113	71	q
18	12	DC2	50	32	2	82	52	R	114	72	r
19	13	DC3	51	33	3	83	53	S	115	73	s
20	14	DC4	52	34	4	84	54	T	116	74	5
21	15	NAK	53	35	5	85	55	U	117	75	u
22	16	SYN	54	36	6	86	56	V	118	76	v
23	17	ETB	55	37	7	87	57	W	119	77	w
24	18	CAN	56	38	8	88	58	X	120	78	x
25	19	EM	57	39	9	89	59	Y	121	79	y
26	1A	SUB	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC	59	3B	;	91	5B	[	123	7B	{
28	1C	FS	60	3C	<	92	5C	\	124	7C	¦
29	1D	GS	61	3D	=	93	5D	]	125	7D	}
30	1E	RS	62	3E	>	94	5E	^	126	7E	~
31	1F	US	63	3F	?	95	5F	-	127	7F	DEL

NOTE:

(1) Simboli delle colonne: DEC = Decimale; HEX = Esadecimale; CHR = Carattere.
(2) I caratteri della colonna HEX che vanno da 0 a 1F sono funzioni di controllo.

viene campionato ad una frequenza che è maggiore del doppio della più elevata frequenza contenuta nel segnale, i campioni forniranno tutte le informazioni contenute nel segnale originale. Se la massima frequenza dei segnali che sono presenti in un sistema di controllo è di 100 hertz, allora il campionamento per la conversione A/D deve essere almeno alla frequenza di 200 volte al secondo.

che se un segnale viene campionato ad una frequenza che sia almeno il doppio della sua più alta frequenza, esso conterrà tutte le informazioni del segnale originale.

Il convertitore A/D

I principi fondamentali di un convertitore A/D sono mostrati in *figura 1.21a*. L'ingresso è un segnale che varia con continuità e che viene campionato a intervalli di tempo specifici determinati dalla frequenza del campionamento. Il valore della tensione misurata viene convertito in un codice digitale che è unico per quel valore. In

Un convertitore A/D trasforma i campionamenti di un ingresso analogico presi a

Figura 1.20. Campionamento di un segnale analogico *(da: N.M. Schmitt ed R.F. Farwell, Understanding Electronic Control of Automation Systems, Texas Instruments Inc., 1983)*

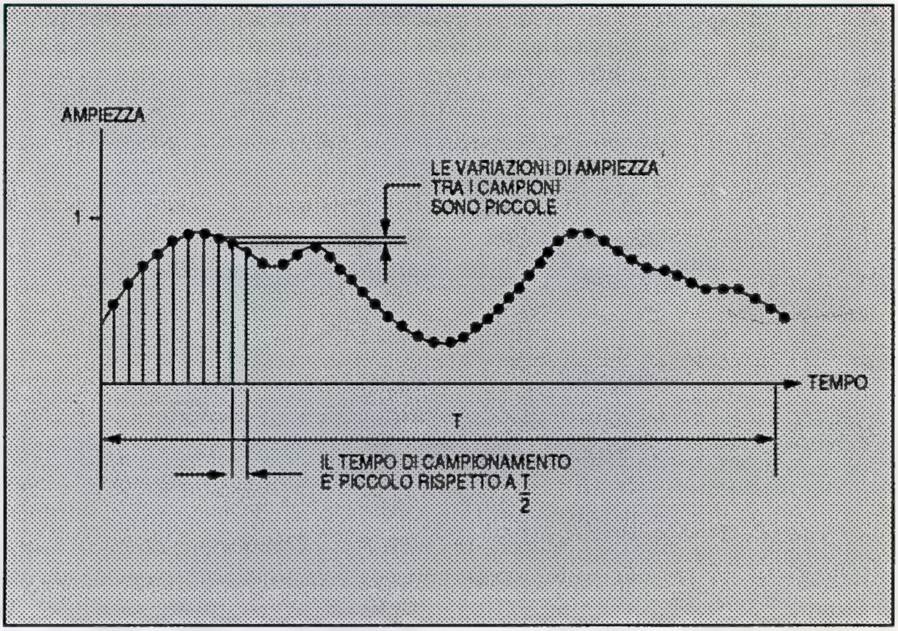
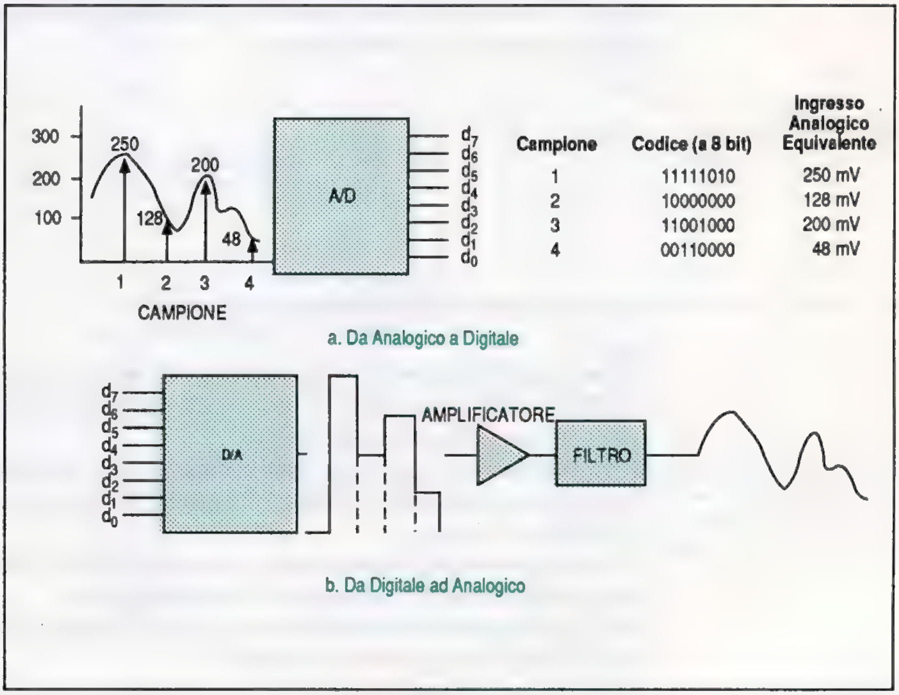

AMPIEZZA
1
LE VARIAZIONI DI AMPIEZZA TRA I CAMPIONI SONO PICCOLE
TEMPO
T
IL TEMPO DI CAMPIONAMENTO È PICCOLO RISPETTO A $\frac{T}{2}$

Figura 1.21. Conversioni dei segnali *(da: J.L.Fike e G.E.Friend, Understanding Telephone Electronics, Texas Instruments Inc., 1983)*


300
200
100
250
200
128
48
1
2
3
4
CAMPIONE
A/D
d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0
D/A
AMPLIFICATORE
FILTRO

Campione	Codice (a 8 bit)	Ingresso Analogico Equivalente
1	11111010	250 mV
2	10000000	128 mV
3	11001000	200 mV
4	00110000	48 mV

a. Da Analogico a Digitale

b. Da Digitale ad Analogico

specifici intervalli di tempo in un unico codice digitale che rappresenta i livelli d'ingresso.

figura 1.21a viene utilizzato un codice a 8 bit. Tutti i bit del codice vengono presentati dal convertitore nello stesso istante (in parallelo) ogniqualvolta l'ingresso viene misurato e il codice rappresenta il valore del segnale di ingresso nell'istante del campionamento. I codici vengono inviati ad altri circuiti digitali del sistema o in parallelo (ogni bit lungo un filo separato dagli altri) o in serie (i bit vengono inviati l'uno dopo l'altro sequenzialmente lungo una sola coppia di fili). Dovrebbe essere evidente che se il codice è un codice a n bit e se il campionamento avviene alla frequenza di s hertz, allora la velocità di trasmissione espressa in bit al secondo è n x s qualora il segnale sia inviato in serie lungo una linea.

Convertitore D/A

Il convertitore D/A utilizza un codice digitale che rappresenta un'uscita analogica per ricostruire il segnale analogico.

Nel convertitore D/A, l'ingresso è uguale o simile al codice in parallelo che formava l'uscita del covertitore A/D. (Se il codice è espresso in forma seriale deve essere convertito nella forma in parallelo per poter essere utilizzato dal convertitore D/A). Il convertitore D/A fornisce un livello di tensione corrispondente al codice di input come mostrato in *figura 1.21b.* Durante il periodo del campionamento il livello della tensione rimane costante; perciò l'uscita presenta livelli di tensione a step (gradino). L'uscita assume una forma molto simile all'originale, variabile con continuità, se viene fatta passare attraverso un amplificatore ed un filtro come mostrato in *figura 1.21b.* Più numerosi sono i campioni A/D rilevati, più fedelmente il segnale filtrato D/A riproduce il segnale originale.

CHE COSA ABBIAMO IMPARATO?

1. I componenti fondamentali dei sistemi digitali sono gli invertitori, le porte combinatorie (AND, OR, NAND, NOR e XOR) e gli elementi per la memorizzazione di un singolo bit chiamati flip-flop.
2. I flip-flop possono essere classificati come asincroni (ingresso senza clock) o sincroni (la memorizzazione avviene solo quando il segnale del clock effettua una transizione di livello).
3. I segnali binari 0 ed 1 possono essere usati come uscita di una singola porta o possono essere trovati in combinazioni dette codici binari.
4. I codici binari possono essere inviati in serie o in parallelo da una parte all'altra del sistema o da un sistema ad un altro.
5. I codici binari possono rappresentare valori numerici (con o senza segno), caratteri alfanumerici o comandi e istruzioni del sistema.
6. Il codice esadecimale viene spesso utilizzato dagli operatori quando lavorano con sistemi digitali binari poiché è più semplice ed è meno probabile la generazione

Quiz del Capitolo 1

1. Se nel circuito di *figura 1.1a* i valori di R_1 ed R_2 vengono sostituiti rispettivamente con 10 e 30 ohm, che valore indicherà il voltmetro?
a. 2,50
b. 5,00
c. 7,50
d. 8,00

2. Nelle stesse condizioni del problema 1, che valore indicherà l'amperometro?
a. 0,25
b. 0,50
c. 0,75

3. Se una tensione sinusoidale ha un valore di picco di 173,2 volt, quale sarà il valore rms?
a. 100
b. 122,45
c. 127,15
d. 173,20

4. Se la sinusoide del problema 3 ha un periodo di 2,5 millisecondi, quale sarà la sua frequenza in hertz?
a. 2500
b. 400
c. 250
d. 100

5. Se una porta OR Esclusivo ha come ingressi 0 ed 1, quale sarà l'uscita?
a. 0
b. 1

6. Se il codice binario 10100110 rappresenta un numero decimale assoluto, qual è il numero decimale equivalente al codice?
a. 140
b. 106
c. 156
d. 166

7. Qual è l'equivalente binario del numero decimale 102?
a. 10010011
b. 01100011
c. 01101001
d. 01100110

8. Qual è l'equivalente decimale del numero esadecimale 3E8?
a. 576
b. 1000
c. 1025
d. 8350

9. Quale numero decimale rappresenta il codice binario con segno (complemento a 2) 10100100?
a. -92
b. -166
c. 92
d. 166

10. Si trasformi il numero decimale 1000 nel suo equivalente binario.
a. 001111101000
b. 000100000000
c. 000000001000
d. 100011100011

11. Si trasformi il numero decimale -32 nel suo equivalente binario (nella forma complemento a 2).
a. 00010000
b. 10010000
c. 00110010
d. 11100000

12. Si trovi l'equivalente decimale del seguente codice BCD: 01100111.
a. -67
b. 103
c. 67
d. 76

13. Scrivere il numero decimale 80 come codice BCD.
a. 01010000
b. 1000000
c. 10000000
d. 00000101

14. Se una resistenza di 10 ohm viene percorsa da una corrente sinusoidale con ampiezza di picco di 14,14 ampere, quanta potenza in watt viene dissipata dalla resistenza?
a. 100
b. 141,4
c. 1000
d. 2000

Funzioni dei sistemi digitali

SCOPO DEL CAPITOLO

Nel capitolo precedente sono stati passati in rassegna alcuni tra i componenti e i segnali digitali fondamentali. Singolarmente i segnali, i codici e i componenti digitali servono a ben poco a meno che essi non vengano combinati in maniera appropriata per svolgere un'operazione di cui si ha bisogno. Una particolare disposizione di porte logiche digitali e di elementi di memoria fondamentali che esegua determinati compiti forma un sistema digitale.

Il funzionamento complessivo di un sistema digitale è ciò che più interessa al progettista del sistema, a chi utilizza il sistema e a chi è preposto all'eliminazione dei guasti del sistema stesso. È necessario capire il funzionamento specifico del sistema per scoprire ed eliminare la causa di un difetto riscontrato nel sistema stesso. Fortunatamente, tutti i sistemi digitali hanno in comune alcune funzioni fondamentali che possono essere facilmente comprese. Queste saranno considerate in questo capitolo.

FUNZIONI DIGITALI GENERALI

Tutti i sistemi digitali contengono le funzioni fondamentali d'ingresso, di uscita e di elaborazione, come mostra la *figura 2.1*. La memoria viene usata in molti, ma non in tutti, i sistemi digitali.

Ingresso

La funzione d'ingresso fornisce uno strumento agli operatori o agli altri sistemi per inserire informazioni e comandi in un sistema digitale.

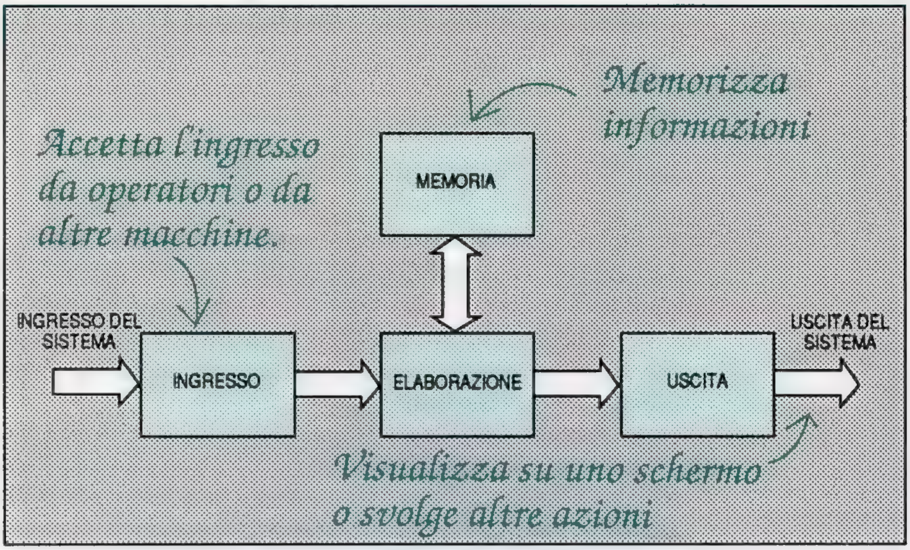
La funzione di ingresso fornisce il modo per trasmettere comandi e informazioni al sistema. L'informazione richiesta da un sistema dipende dallo scopo del sistema stesso. L'informazione può provenire da operatori (comandi) o può provenire da sensori che convertono grandezze fisiche come la temperatura, la posizione e la velocità, in segnali elettrici che possono essere elaborati. Lo scopo della funzione di ingresso è di fornire un'interfaccia adeguata tra il mondo esterno e la funzione di elaborazione. Quest'interfaccia può contenere segnali di condizionamento, memorizzazione temporanea e/o smistamento.

Elaborazione

La funzione di elaborazione svolge operazioni sugli ingressi del sistema per

La funzione di elaborazione è la principale ragione per cui un particolare sistema esiste. Nella sua forma più semplice, la funzione di elaborazione agisce semplicemente come un interruttore o un dispositivo di instradamento tra l'ingresso e l'uscita. Comunque, per la maggior parte del tempo, la funzione di elaborazione agisce sugli ingressi per produrre nuove uscite.

Figura 2.1. Funzioni digitali principali


Memorizza informazioni
Accetta l'ingresso da operatori o da altre macchine.
MEMORIA
INGRESSO DEL SISTEMA
INGRESSO
ELABORAZIONE
USCITA
USCITA DEL SISTEMA
Visualizza su uno schermo o svolge altre azioni

produrre nuove uscite.

Per esempio, una calcolatrice portatile accetta numeri e comandi che vengono immessi da un operatore attraverso la tastiera. La funzione di elaborazione opera sui numeri a seconda del comando impartito. Per esempio, se si preme il tasto +, si comanda al processore di sommare i numeri ed inviare i risultati alla funzione di uscita perché appaiano all'operatore sul display. Ci sono molti esempi di funzione di elaborazione in quanto ci sono molti tipi di sistemi digitali. Generalmente tutte le principali caratteristiche operative e decisionali fondamentali del sistema sono incorporate nei circuiti che formano la funzione di elaborazione. In alcuni tipi di strutture di un sistema, la funzione di elaborazione può essere distribuita attraverso tutto il sistema. Se la struttura è quella di un computer, ci sarà un circuito centralizzato (o un gruppo di circuiti) chiamato processore (o elaboratore).

Uscita

La funzione d'uscita del sistema visualizza su di uno schermo i risultati per gli operatori e/o svolge o controlla alcune operazioni.

Dopo che l'informazione è stata elaborata, la funzione di uscita invia i risultati verso il mondo esterno. La funzione di uscita, come quella di ingresso, può contenere segnali di condizionamento, memorizzazione temporanea e/o smistamento. Solitamente i risultati in uscita vengono sia mostrati sul display (per gli operatori) sia utilizzati per controllare i dispositivi fisici esterni. Perciò i segnali di controllo possono avviare un motore o accendere una luce, regolare la velocità di un motore o l'intensità di una lampada, arrestare un motore o spegnere una lampada.

Memoria

La funzione di memoria non è necessaria in tutti i sistemi, ma è usata in quelli più complessi. La memoria è necessaria quando la funzione di elaborazione deve ricordare un'informazione per utilizzarla in seguito.

La memoria è

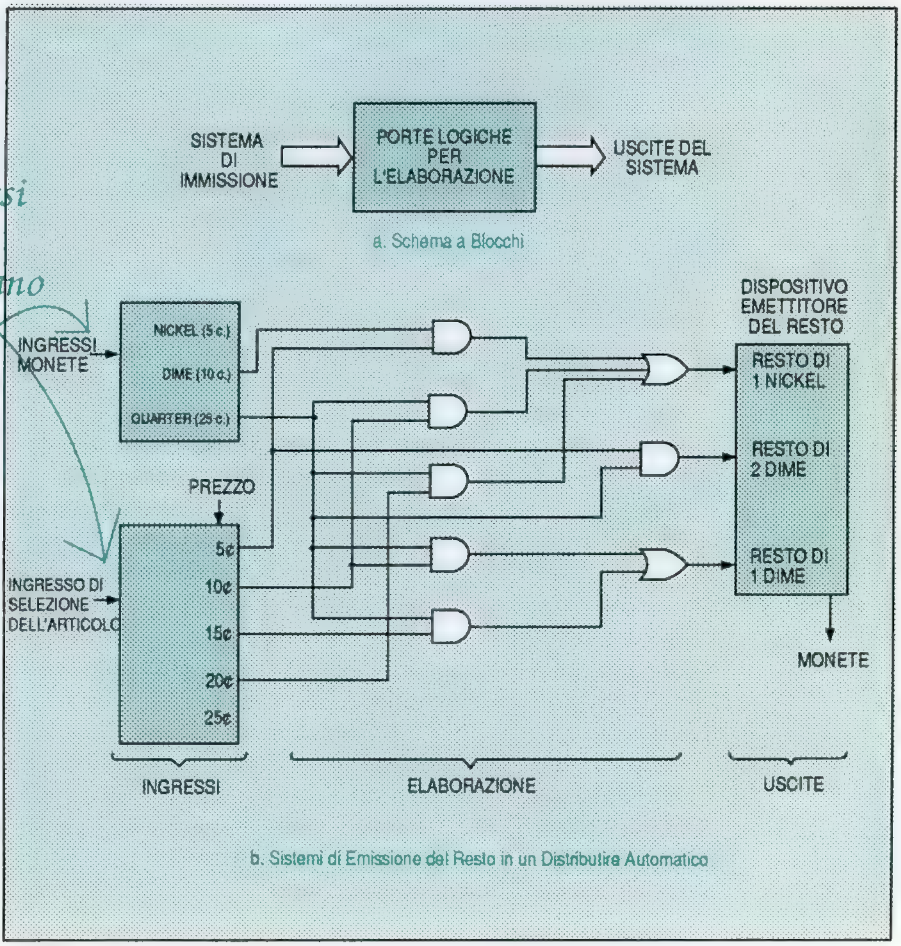
Per esempio, il sistema di controllo di un ascensore deve sapere dove si trova la cal'immagazzinamento d'informazioni, dati o istruzioni da parte del sistema da utilizzarsi durante lo svolgersi dell'elaborazione.

bina rispetto al piano a cui è chiamata per determinare se farla salire o scendere. Un circuito di memoria tiene nota della posizione della cabina in ogni momento. Così, il processore può ottenere la posizione della cabina dell'ascensore dalla memoria e il piano dal quale proviene la chiamata dalla funzione di ingresso; può quindi calcolare in che direzione e di quanto la cabina debba spostarsi. I risultati di questo calcolo passano attraverso la funzione di uscita per controllare la direzione e la velocità del motore del paranco.

Nel sistema di un calcolatore la funzione di memoria è richiesta per immagazzinare informazioni o dati durante l'elaborazione ed anche per immagazzinare la sequenza delle operazioni (istruzioni o programmi) che deve essere svolta sulla base delle informazioni. Fondamentalmente, la memorizzazione è richiesta in tutti quei sistemi il cui comportamento dipende da un comportamento precedente. Il fatto che la memoria venga o non venga utilizzata in un dato sistema determina il tipo di sistema. I tipi fondamentali di sistemi verranno considerati nel seguente paragrafo.

Figura 2.2. Sistema logico combinatorio

Gli ingressi presenti determinano le uscite


SISTEMA DI IMMISSIONE
PORTE LOGICHE PER L'ELABORAZIONE
USCITE DEL SISTEMA
INGRESSI MONETE
NICKEL (5 c.)
DIME (10 c.)
QUARTER (25 c.)
PREZZO
INGRESSO DI SELEZIONE DELL'ARTICOLO
5¢
10¢
15¢
20¢
25¢
DISPOSITIVO EMETTITORE DEL RESTO
RESTO DI 1 NICKEL
RESTO DI 2 DIME
RESTO DI 1 DIME
MONETE
INGRESSI
ELABORAZIONE
USCITE

a. Schema a Blocchi

b. Sistemi di Emissione del Resto in un Distributore Automatico

TIPI DI SISTEMI DIGITALI

I sistemi digitali possono essere classificati in due tipi generali: sequenziali e combinatori, a seconda che venga impiegata o meno la funzione di memorizzazione.

Sistemi combinatori

I sistemi che non richiedono l'immagazzinamento di informazioni (memoria) vengono chiamati sistemi digitali combinatori. Un sistema di questo tipo è mostrato sotto forma di schema a blocchi in *figura 2.2a*. In un sistema logico combinatorio, una volta che siano disponibili degli ingressi, le uscite possono essere determinate senza dover fare riferimento a condizioni di ingresso precedenti. In altre parole, le uscite del sistema sono in funzione solamente delle condizioni di ingresso attuali.

Un semplice distributore automatico

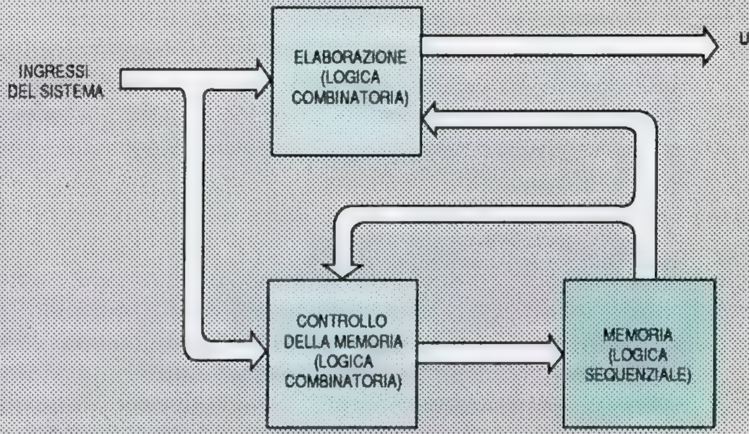
I sistemi combinatori non richiedono una memoria in quanto le loro uscite emettono un valore definito e prefissato per una data combinazione di ingressi.

Il sistema digitale combinatorio mostrato in *figura 2.2b* consiste solo di porte AND e OR che forniscono lo schema logico per un semplice distributore automatico. Una serie di ingressi indica il valore della moneta inserita nel distributore automatico, mentre un'altra serie di ingressi indica il costo dell'articolo da acquistare. Per semplicità, si assuma che possa venire inserita una sola moneta, un nickel (5 cent) o un dime (10 cent) o un quarter (25 cent), per l'acquisto e che il costo di ogni articolo disponibile sia di 5, 10, 20 o 25 cent. Ovviamente non c'è resto se un nickel viene utilizzato per un acquisto di 5 cent, un dime per un articolo di 10 cent o un quarter per un articolo di 25 cent.
Occorre dare un resto in nickel quando viene inserito un dime per acquistare un articolo che costa un nickel o quando si inserisce un quarter per articoli che costano sia 10 che 20 cent. Si dà un resto in dime quando viene inserito un quarter per acquistare un articolo che costa o 10 o 15 cent. Se viene inserito un quarter per comprare un articolo da 5 cent devono essere restituite in resto due monete da 10 cent. La semplice serie di porte AND e OR mostrata in *figura 2.2b* fornisce lo schema logico per effettuare queste operazioni di resto. Il lettore può verificarlo considerando tutte le condizioni che richiedono l'emissione di un resto. Un altro punto su cui si dovrebbe porre l'attenzione è che i valori reali delle monete non devono essere utilizzati. Al loro posto può essere utilizzato un codice digitale appropriato che rappresenti i tre possibili tipi di monete.
Naturalmente un distributore automatico reale è più complesso, considerando il fatto che il costo di un articolo può essere superiore ai 25 cent e considerando che possono essere inserite differenti combinazioni di monete. Comunque, se possono essere inserite più monete, il sistema di controllo digitale deve ricordare cosa è stato inserito allo scopo di dare un resto esatto. Perciò viene richiesta una funzione di memoria e il sistema non è più un sistema combinatorio, ma un sistema digitale sequenziale.

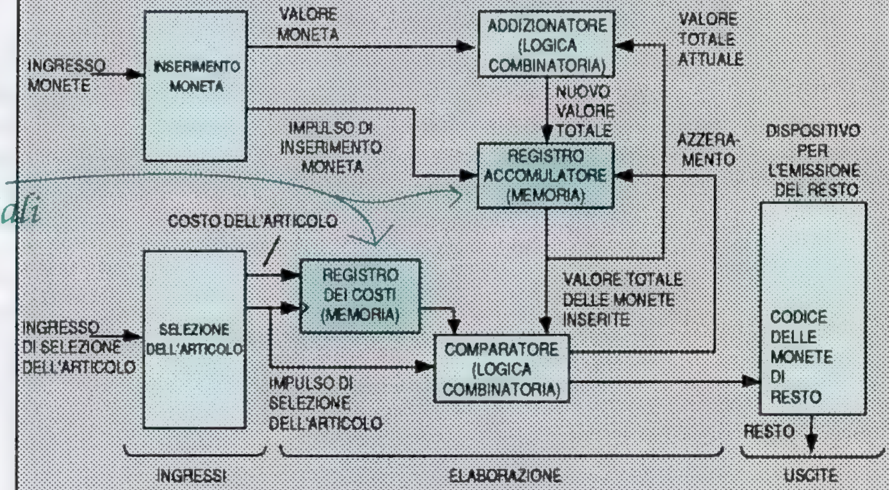
Sistemi sequenziali

Le principali funzioni in un sistema digitale sequenziale sono mostrate sotto forma

**Figura 2.3.
Sistema logico sequenziale**


INGRESSI DEL SISTEMA
ELABORAZIONE (LOGICA COMBINATORIA)
USCITE DEL SISTEMA
CONTROLLO DELLA MEMORIA (LOGICA COMBINATORIA)
MEMORIA (LOGICA SEQUENZIALE)

a. Schema a Blocchi di un Sistema sequenziale


INGRESSO MONETE
INSERIMENTO MONETA
VALORE MONETA
IMPULSO DI INSERIMENTO MONETA
ADDIZIONATORE (LOGICA COMBINATORIA)
VALORE TOTALE ATTUALE
NUOVO VALORE TOTALE
REGISTRO ACCOMULATORE (MEMORIA)
AZZERA-MENTO
DISPOSITIVO PER L'EMISSIONE DEL RESTO
COSTO DELL'ARTICOLO
INGRESSO DI SELEZIONE DELL'ARTICOLO
SELEZIONE DELL'ARTICOLO
REGISTRO DEI COSTI (MEMORIA)
VALORE TOTALE DELLE MONETE INSERITE
COMPARATORE (LOGICA COMBINATORIA)
IMPULSO DI SELEZIONE DELL'ARTICOLO
CODICE DELLE MONETE DI RESTO
RESTO
INGRESSI
ELABORAZIONE
USCITE

Circuiti Sequenziali

b. Dispositivo di emissione del Resto in un Distributore Automatico più Complesso

I sistemi sequenziali hanno una memoria che immagazzina informazioni sia inizialmente che da operazioni già svolte in caso si debbano utilizzare per operazioni presenti.

di schema a blocchi in *figura 2.3a*. Il sistema digitale sequenziale consiste di uno o più sottosistemi combinatori e di un sottosistema sequenziale (memoria). Generalmente la memoria è costituita da un gruppo di flip-flop per sistemi che non devono eseguire un calcolo. In un sistema sequenziale la memoria tiene nota delle operazioni eseguite nell'immediato passato in quanto le operazioni future dipendono da questa informazione.

In *figura 2.3a* tra i componenti di sistema sequenziale è incluso un blocco logico combinatorio che elabora gli ingressi attuali e le uscite della memoria per generare i segnali di uscita del sistema. Un altro blocco logico combinatorio controlla gli ingressi di memoria (i valori successivi da memorizzare in base agli ingressi attuali) e le uscite di memoria.

Distributore automatico complesso

I sitemi digitali complessi contengono sia circuiti combinatori che circuiti sequenziali.

La *figura 2.3b* mostra un esempio di un sistema sequenziale. Esso è basato sull'esempio di *figura 2.2*, ma questa volta la reale configurazione delle porte logiche non è mostrata. I differenti blocchi contengono numerose porte. In questo esempio la macchina è in grado di ricevere più di un solo tipo di monete alla volta e il costo degli articoli a disposizione va dai 5 ai 100 cent, in incrementi di 5 cent. I circuiti di memoria devono registrare quanto denaro è stato ricevuto e quindi il costo dell'articolo deve essere sottratto da questa somma per determinare il resto che deve essere ritornato all'acquirente. Si potrebbe anche chiedere alla memoria di registrare quante monete della stessa pezzatura sono state rese, qualora ne vengano rese più d'una.

Il rilevatore di ingresso della moneta identifica il tipo di moneta che è stato inserito e attraverso un codice digitale invia il valore a uno degli ingressi dell'addizionatore. Inoltre ciascuna moneta inserita fa sì che il rilevatore fornisca un impulso che porta il valore dal registro accumulatore (che inizialmente è 0) all'altro ingresso dell'addizionatore. L'addizionatore somma i due valori e memorizza la somma nel registro dell'accumulatore. Questo procedimento viene ripetuto per ogni moneta inserita. (Il registro è una funzione di memoria).

Quando viene fatta la selezione dell'oggetto, l'impulso di selezione inserisce il costo dell'oggetto nel registro dei prezzi. L'impulso di selezione dell'oggetto porta anche il comparatore logico a sottrarre il costo dell'oggetto dalla somma totale di denaro ricevuto. Se la somma ricevuta è maggiore del costo dell'oggetto il comparatore logico determina la combinazione di monete differenti da restituire e le sue uscite indicano il numero di nickel, dime e quarter che devono essere resi. Questi codici numerici vengono inviati al meccanismo che restituisce le monete il quale, a sua volta, emette il numero richiesto di ogni tipo diverso di moneta nel vano apposito per la restituzione. Un segnale proveniente dal comparatore poi cancella (azzera) il registro dell'accumulatore per prepararlo alla selezione successiva.

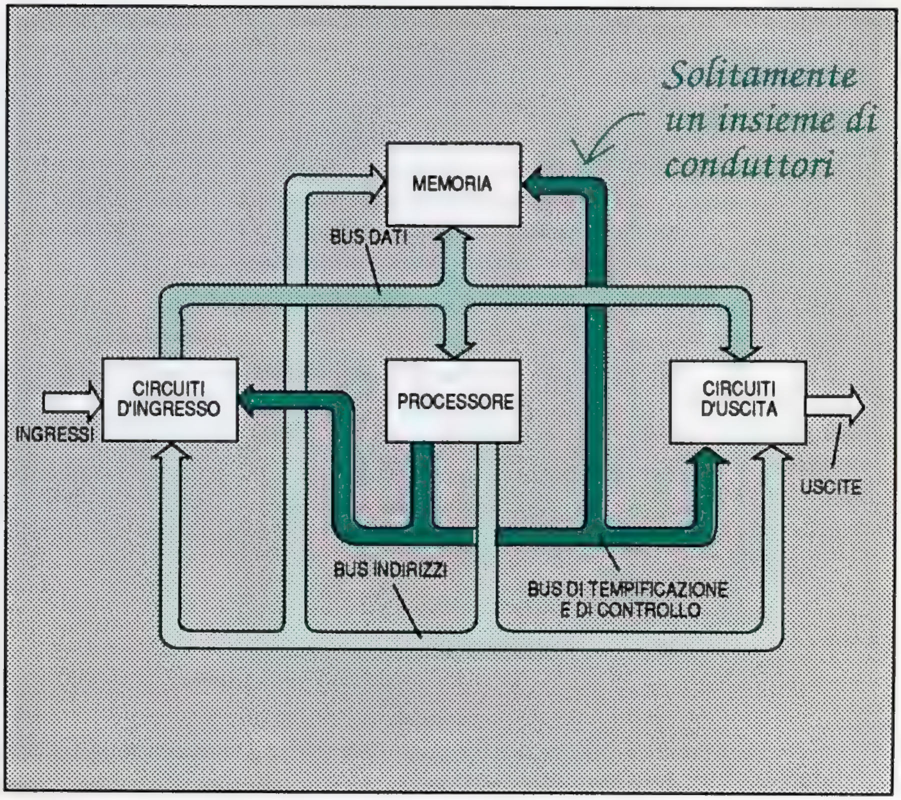
Il meccanismo che restituisce le monete cosi come quello di ingresso delle monete stesse possono essere sistemi logici digitali, combinatori o sequenziali. In questo esempio, tuttavia, essi vengono considerati semplicemente come componenti, così come sono componenti i registri e le porte AND e OR. Il sistema sequenziale di *figura 2.3* è più complicato del sistema combinatorio di *figura 2.2*, ma può eseguire operazioni più complesse.

Strutture

I sistemi digitali complessi possono essere formati da un solo circuito integrato (un esempio è dato dal microprocessore o dal microcomputer) o possono essere costruiti come combinazione di funzioni SSI, MSI e LSI.

Poiché solo i sistemi più semplici possono essere costruiti utilizzando solamente la logica combinatoria (porte), la maggior parte dei sistemi presenta anche una logica sequenziale (memoria). Ci sono molti tipi differenti di strutture di un sistema sequenziale. Quella illustrata in *figura 2.3b* potrebbe utilizzare porte di un circuito integrato (IC) a piccola scala di integrazione (SSI, da Small Scale Integration) e registri di un circuito integrato a media scala di integrazione (MSI, da Medium Scale Integration). Alcuni sistemi includono memorie di circuiti integrati a grande scala di integrazione (LSI da Large Scale Integration) ed altri consistono solo di componenti SSI quali porte logiche e flip-flop. Sistemi sequenziali complessi sono disponibili sotto forma di un singolo circuito integrato in scala di integrazione molto grande (VLSI da Very Large Scale Integration). Questi tipi complessi di sistemi sequenziali generalmente vengono costruiti utilizzando la struttura di un computer e vengono chamati microcomputer. Essi vengono usati in svariate operazioni di controllo e costituiscono il cuore dei personal computer e dei piccoli calcolatori. Incrementando la capacità e la velocità di questi sistemi, essi diventano minicomputer e persino grandi computer di uso generale. Poiché in futuro la struttura del computer sarà la più comune, il resto di questo capitolo e gran parte di questo libro saranno dedicati a questo tipo di sistema digitale.

Figura 2.4. Struttura di un calcolatore



FUNZIONI DEL COMPUTER

Un computer è un sistema digitale sequenziale complesso che contiene funzioni di memoria, di ingresso, di elaborazione e di uscita.

Un computer è un sistema digitale sequenziale complesso che include tutte le funzioni fondamentali mostrate in *figura 2.1*; comunque le funzioni sono collegate fra loro come mostrato in *figura 2.4*. Queste funzioni possono essere svolte da quattro o più differenti circuiti integrati (IC). Se il processore è un IC separato, esso viene chiamato microprocessore. Se tutte e quattro le funzioni sono collocate in un singolo IC, esso viene chiamato microcomputer.

Il circuito integrato

Un circuito integrato digitale è un circuito elettronico completo collocato fisicamente in un piccolo contenitore, plastico o ceramico. Esso svolge le operazioni ad alta velocità, è economico e molto affidabile.

Un circuito integrato è un circuito con una o più funzioni complete realizzato su una singola piastrina (detta chip) di materiale allo stato solido (silicio) e sigillato in un piccolo contenitore di plastica, metallo o ceramica, con conduttori metallici che servono per il collegamento con il mondo esterno. Alcuni circuiti analogici, come gli amplificatori operazionali e quasi tutti i circuiti digitali, sono disponibili sotto forma di IC. Questi circuiti sono altamente affidabili, a basso costo, hanno un basso consumo di potenza e sono piccoli, il che semplifica il progetto, la costruzione e la verifica dei moderni sistemi elettronici.

Il concetto di bus

Normalmente un bus è un insieme di fili o di conduttori stampati che trasportano dati digitali in parallelo. I bus utilizzati per collegare un computer sono: il bus dati, il bus di tempificazione e di controllo e il bus indirizzi.

I tre tipi di bus che mettono in connessione tra loro le parti funzionali sono mostrati in *figura 2.4*. Nonostante un bus possa essere un singolo conduttore, così come lo è un bus di massa, il termine viene abitualmente riferito a un gruppo di conduttori di circuito stampato o di fili attraverso cui vengono inviati segnali digitali in parallelo. Nei microcomputer il bus ha solitamente 8 o 16 conduttori e viene perciò detto ad 8 (o 16) bit perché i conduttori sono collocati fianco a fianco. Il conduttore piatto e largo costituito da cavi isolati viene chiamato cavo piatto.

Bus dati

Il bus dati consente il trasferimento di dati (le informazioni alfanumeriche del sistema) tra il processore e le altre funzioni: memoria, ingresso e uscita. Il bus dati è a due vie, cosicché le informazioni possono essere trasferite da e verso il processore, ma in una sola direzione allo stesso istante. Il processore controlla quali funzioni può utilizzare il bus dati e quando può farlo. Questi segnali di tempificazione e di controllo vengono inviati dal processore al bus di controllo e tempificazione che è monodirezionale.

Bus di controllo e tempificazione

I segnali su questo bus determinano quando i dispositivi di memoria, di ingresso o di uscita possono trasferire dati al processore o riceverne. Il trasferimento al processore di dati provenienti da un ingresso o da una memoria è chiamato operazione di lettura. Il trasferimento dal processore all'uscita o alla memoria viene detto operazione di scrittura.

Bus indirizzi

Il bus monodirezionale di indirizzamento viene usato per inviare codici di locazione dal processore alle altre funzioni. L'indirizzo indica quale posizione nei sottosistemi di ingresso, di uscita o di memoria deve inviare o ricevere dati dal processore. Il processore genera un indirizzo o lo ottiene da un'istruzione, e colloca questo indirizzo sul bus indirizzi. Quindi esso invia segnali di tempificazione e di controllo sul bus di tempificazione e di controllo per attivare i sottosistemi d'ingresso o di uscita o di memoria e per generare o un'operazione di lettura dalla posizione indirizzata o un'operazione di scrittura nella posizione indirizzata.

I segnali sui tre bus esistono solo per un breve periodo di tempo.

Solitamente i segnali di indirizzamento, tempificazione e controllo restano sui bus per circa 250 - 500 nanosecondi. Il trasferimento di dati tra il processore e il sottosistema selezionato deve avvenire durante questo intervallo di tempo. Naturalmente è necessaria una strumentazione speciale per poter osservare lo svolgersi di eventi tanto veloci e per scoprire se le configurazioni appropriate del segnale sono avvenute nell'istante appropriato. Tale strumentazione per il test verrà trattata nei capitoli seguenti.

Istruzioni

Il programma di un computer immagazzinato nella memoria è una sequenza d'istruzioni passo a passo che definiscono le operazioni che il sistema deve svolgere per eseguire un compito dato.

Oltre ad immagazzinare dati la memoria di un computer registra codici di istruzione dal processore. I codici di istruzione devono essere codici binari inclusi nella serie di istruzioni utilizzate per il particolare processore. La serie di istruzioni viene fissata dal progetto; perciò solo le istruzioni nella serie saranno riconosciute dal processore come istruzioni o operazioni che esso è in grado di eseguire. Leggere ed eseguire queste istruzioni è il ruolo del processore nel sistema di un computer.

Le istruzioni indicano passo per passo le operazioni che il sistema deve eseguire per svolgere un dato compito. Queste istruzioni sono il programma del computer e definiscono la funzione fondamentale del sistema. Normalmente le istruzioni sono immagazzinate nella memoria del sistema in posizione sequenziale: cioè in posizioni che vengono l'una dopo l'altra. Quando il sistema viene acceso per la prima volta il processore comincia da una posizione predeterminata a prelevare la prima istruzione del programma dalla memoria e contemporaneamente la memorizza nel registro istruzioni all'interno del processore. L'istruzione è decodificata da circuiti combinatori per determinare di che istruzione si tratta e quale operazione deve essere eseguita. Quindi l'istruzione successiva viene prelevata dalla memoria e posta nel registro istruzioni, e così via. I circuiti sequenziali nel processore generano indirizzi, dati e sequenze di segnali di tempificazione per controllare il prelevamento, la codifica e l'esecuzione dell'istruzione.

ALTRI COMPONENTI DIGITALI

Nel primo capitolo sono stati trattati i componenti digitali fondamentali, ma poiché i sistemi digitali divengono più complessi, essi non hanno come componenti soltanto porte logiche e flip-flop. Questo avviene particolarmente nei computer dove ciascuna componente può contenere migliaia di porte logiche e flip-flop in un singolo circuito integrato. Alcuni di questi componenti, come il registro, sono già stati citati, ma ora li esamineremo più dettagliatamente.

Registri

La memoria temporanea all'interno di un sistema basato su microcomputer che solitamente immagazzina da 8 a 16 bit viene fornita da un gruppo di flip-flop, detto registro.

Un registro è un gruppo di flip-flop che fornisce una funzione di memoria per una particolare funzione di elaborazione, come per esempio un registro di ingresso, un registro di uscita, un registro accumulatore e un registro istruzioni. I registri nei microcomputer solitamente possono memorizzare 8 o 16 bit. Poiché ciascun flip-flop può memorizzare un bit, in un registro vengono raggruppati 8 o 16 flip-flop per formare un byte (o una parola).

Solitamente i flip-flop operano tutti contemporaneamente, cosicché tutti i bit vengono trasferiti dentro o fuori dal registro nello stesso istante per operazioni di dati in parallelo. Comunque alcuni registri sono progettati in modo tale da ricevere dati in parallelo e farli uscire in serie. Altri lavorano esattamente al contrario: ricevono dati in serie e li forniscono in parallelo. Per ora consideriamo solo registri che ricevono e forniscono dati in parallelo.

I registri inviano e ricevono dati in parallelo, convertono ingressi paralleli in uscite seriali o viceversa, o forniscono una semplice funzione di tempificazione per il trasporto delle informazioni.

Il funzionamento di ciascun flip-flop preso individualmente è simile a quello del flip-flop tipo D mostrato in *figura 1.12*. La struttura di un registro a 4 bit è mostrata in *figura 2.5*. I terminali di clock dei 4 flip-flop di tipo D sono collegati fra loro e questo collegamento comune è il terminale di clock esterno. Quindi, quando il segnale di clock esterno passa dal livello 0 al livello 1, qualunque sia il valore sull'ingresso del registro questo viene memorizzato nei quattro flip-flop. L'informazione apparirà entro breve tempo sui terminali di uscita dei flip-flop. I registri solitamente vengono costruiti in versioni da 4 o 8 bit. Se viene richiesto un registro a 16 bit, vengono collegati due registri da 8 bit. Il registro mostrato in *figura 2.5* può fornire una memorizzazione temporanea per un codice a 4 bit. In seguito, esso può fungere da dispositivo di uscita per fornire il codice a 4 bit immagazzinato. Oppure il registro può essere usato non per memorizzare, ma semplicemente come un metodo per fornire una porta all'informazione che dal processore va ad un dispositivo di uscita. Per far sì che il registro fornisca il suo contenuto basta dare un segnale di clock al terminale di clock ogniqualvolta il processore è pronto ad emettere dati. Comunque, poiché ci sono molti registri all'interno di un computer, deve esserci un circuito che esamina i segnali di tempificazione e i segnali di indirizzamento che provengono dal processore ed assicura che i dati vadano nel registro giusto e al momento giusto.

Scelta di un registro

La scelta del registro che deve inviare o ricevere dati viene fatta inviando un indirizzo che sceglie il registro, insieme

La struttura fondamentale è mostrata in *figura 2.6*, con il circuito che distribuisce i segnali di clock ai registri di uscita indicato come un decodificatore di indirizzi. I dettagli di un circuito decodificatore saranno considerati fra poco. Per ora, diciamo soltanto che quando il processore vuole far uscire i dati da uno dei registri di uscita, esso colloca l'indirizzo di quel registro di uscita sul bus indirizzi. Nell'esempio di *figura 2.6* viene mostrato un indirizzo ad 8 bit, che può selezionare uno tra 256 registri di uscita differenti. (Il numero 8 con la sbarra attraverso la linea indica che la sin-

Figura 2.5.
Registro a 4 bit

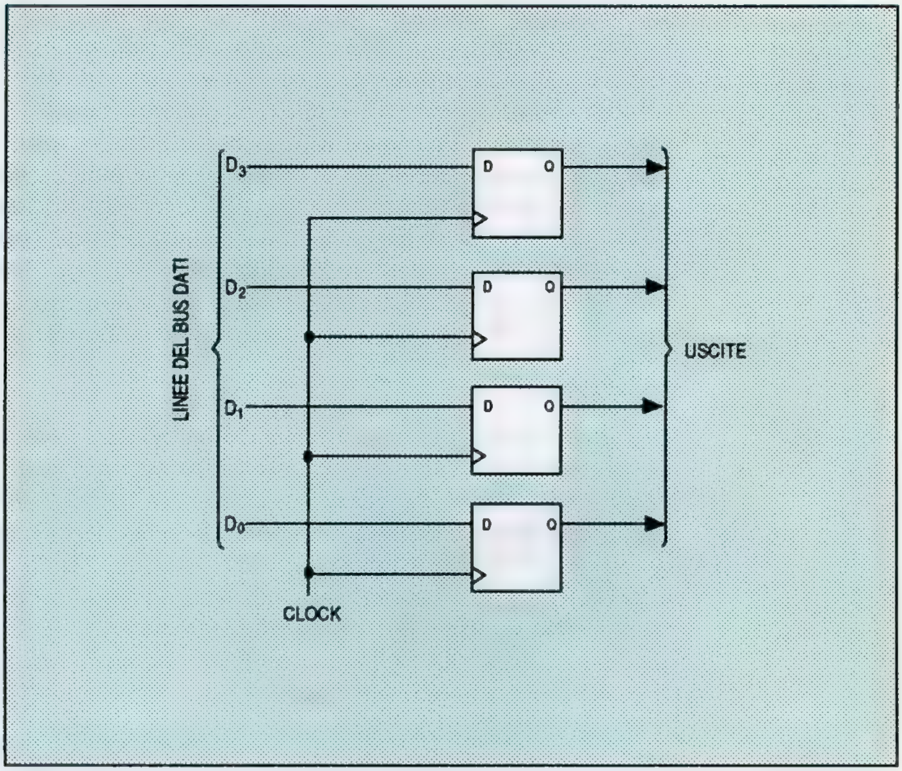
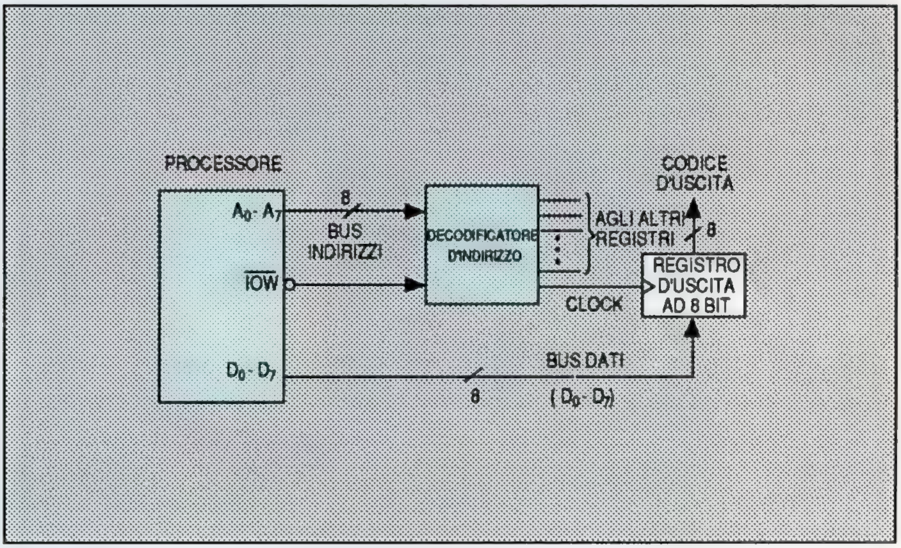

D_3
D_2
D_1
D_0
LINEE DEL BUS DATI
D Q
D Q
D Q
D Q
USCITE
CLOCK

Figura 2.6.
Struttura del registro d'uscita di un processore


PROCESSORE
A_0 - A_7
8
BUS INDIRIZZI
$\overline{IOW}$
D_0 - D_7
DECODIFICATORE D'INDIRIZZO
AGLI ALTRI REGISTRI
CODICE D'USCITA
8
REGISTRO D'USCITA AD 8 BIT
CLOCK
BUS DATI
8
(D_0 - D_7)

con gli appropriati segnali di controllo e di tempificazione.

gola linea rappresenta un bus ad 8 linee). La figura mostra anche che il registro ed il bus dati possono trattare 8 bit di dati digitali in parallelo. Poiché il numero dei bit del bus dati è solitamente lo stesso dei registri interni del processore, si assume che questo processore sia un processore ad 8 bit. Inoltre si suppone che il processore fornisca un segnale di 0, chiamato $\overline{\text{IOW}}$ (abbreviazione per Input Output Write Not), della durata di alcune centinaia di nanosecondi per indicare quando i dati sono in viaggio verso il sottosistema di uscita. I dati dovrebbero essere memorizzati nel registro alla fine di questo intervallo di 0, cioè quando si ha la transizione dal livello 0 al livello 1. Il compito del decodificatore è di decodificare l'indirizzo quando il segnale $\overline{\text{IOW}}$ è attivo ed inviare un impulso al terminale di clock del registro di uscita appropriato. Ora impariamo come lavora il decodificatore.

Decodificatori

Un esempio di decodificatore di linea 3 a 8 mostra come un indirizzo a 3 bit di selezione del registro attiva il clock di uno degli 8 registri d'uscita per selezionare il

In *figura 2.7* è mostrato un tipico circuito decodificatore 3-a-8. Il segnale sul terminale di abilitazione $\overline{E}$ deve essere basso per permettere a qualsiasi uscita di andare a 0 indipendentemente dai segnali agli ingressi di selezione A, B e C. Se $\overline{E}$ viene portato ad 1 anche tutte le uscite del decodificatore saranno ad 1. Quando $\overline{E}$ diventa basso, una e solo una delle 8 uscite diventerà bassa. Quale sia l'uscita che diviene bassa sarà determinato dal codice binario a 3 bit agli ingressi di selezione del decodificatore. Perciò se viene dato il codice a 3 bit 000 mentre $\overline{E}$ è a 0 l'uscita 0 sarà a 0. Se viene dato il codice a 3 bit 111 mentre $\overline{E}$ è a 0 l'uscita 7 sarà a 0.

Ora facciamo riferimento alla *figura 2.8* per notare che $\overline{E}$ diventerà basso quando

Figura 2.7. Collegamento dei piedini di un decodificatore 3 a 8

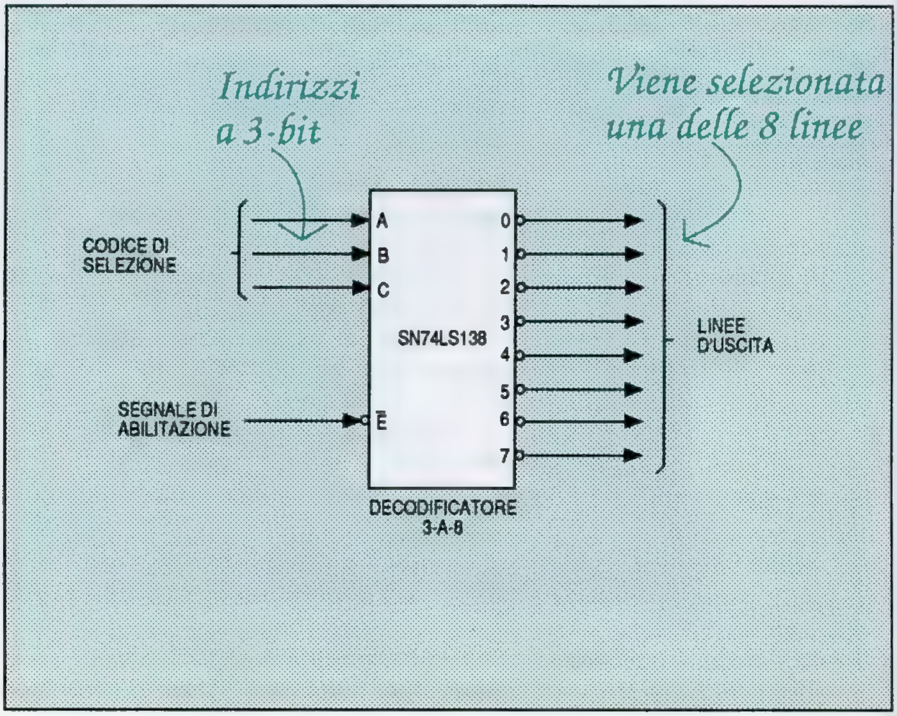
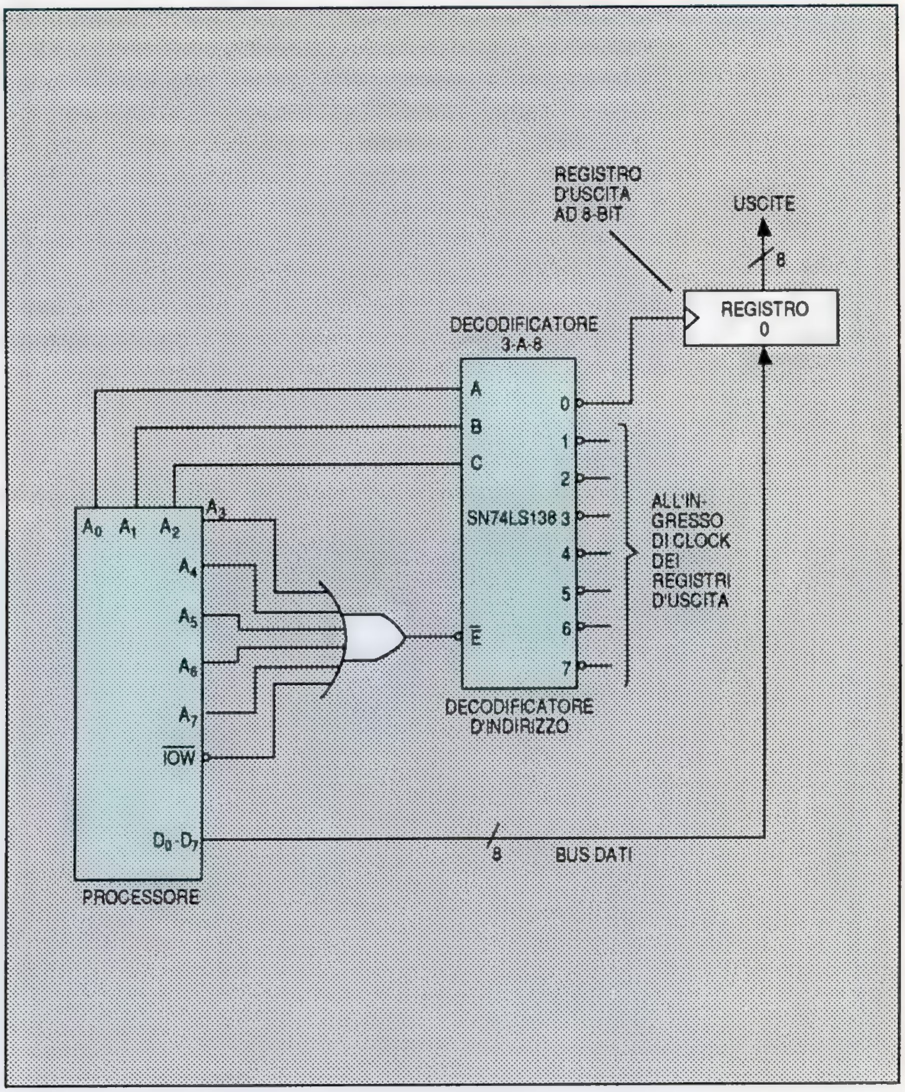


Figura 2.8. Sistema d'uscita di un computer


REGISTRO
D'USCITA
AD 8-BIT
USCITE
8
REGISTRO
0
DECODIFICATORE
3-A-8
A
B
C
0
1
2
SN74LS138 3
4
5
6
7
$\overline{E}$
ALL'INGRESSO DI CLOCK DEI REGISTRI D'USCITA
DECODIFICATORE D'INDIRIZZO
A_0 A_1 A_2 A_3 A_4 A_5 A_6 A_7
$\overline{IOW}$
D_0-D_7
8
BUS DATI
PROCESSORE

registro desiderato.

$\overline{\text{IOW}}$ e tutti gli altri ingressi della porta OR sono bassi. Gli ingressi di selezione dal decodificatore sono collegati con i 3 bit meno significativi del bus indirizzi. I 5 bit più significativi del codice indirizzi e il segnale $\overline{\text{IOW}}$ vengono inviati ad una porta OR la cui uscita viene inviata ad $\overline{\text{E}}$. La linea d'uscita 0 viene inviata al clock di un registro di uscita. Si avrà un'uscita dal registro (e verranno memorizzati nel registro i dati provenienti dal processore) ogniqualvolta tutte le linee di indirizzo sono a 0 mentre $\overline{\text{IOW}}$ è basso. Questo si verifica perché soltanto se tutti gli ingressi alla porta OR sono a 0 l'uscita sarà a 0 (essendo il segnale di ingresso $\overline{\text{E}}$). Perciò l'indirizzo che corrisponde al registro mostrato sarà il codice binario ad 8 bit 0000 0000 che può anche essere espresso come codice esadecimale di 00_{16} . Se vengono aggiunti al

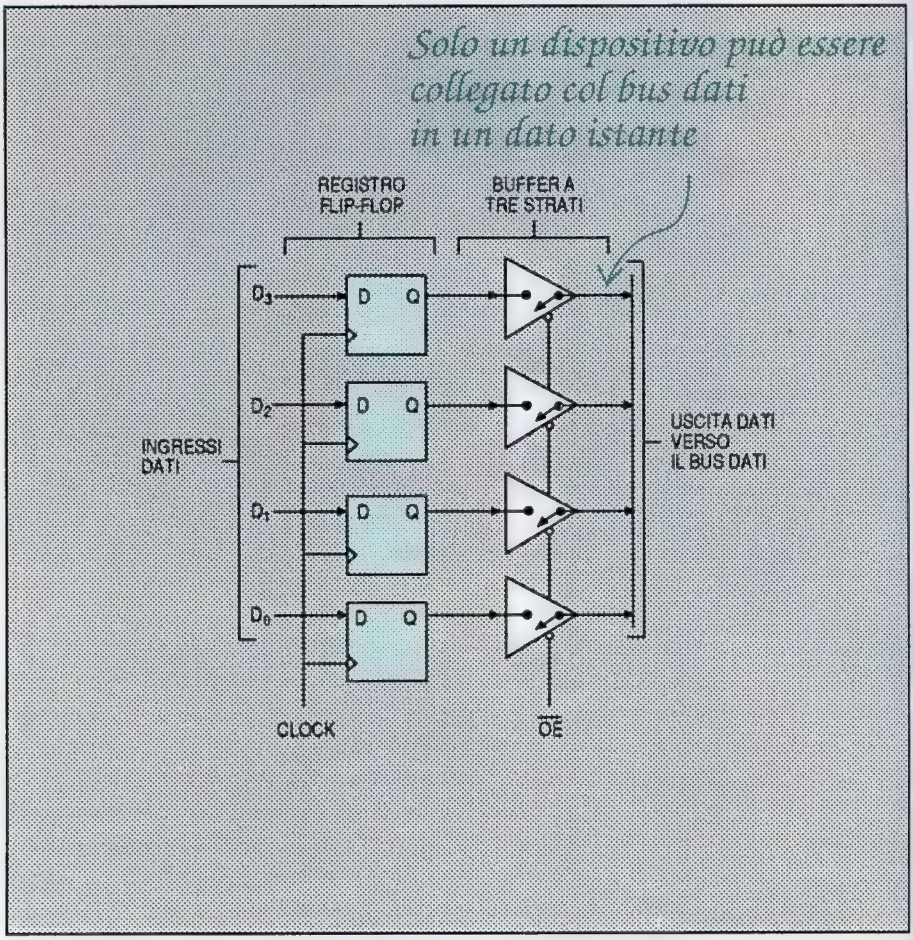
I registri d'ingresso funzionano fondamentalmente come i registri d'uscita; comunque essi richiedono segnali di controllo differenti e un buffer a tre stati sulle loro uscite.

sistema altri 7 registri e le uscite da 1 a 7 del decodificatore sono inviate ai terminali di clock dei registri da 1 a 7, rispettivamente, questi registri aggiuntivi risponderebbero agli indirizzi da 01_{16} a 07_{16} durante un'istruzione di uscita al registro. Utilizzando combinazioni di decodificatori 3-a-8 sarebbe possibile sviluppare segnali di clock per un massimo di 256 registri di uscita (ad indirizzi da 00_{16} a FF_{16}). Naturalmente ogni decodificatore (ne sarebbero necessari 32) dovrebbe rispondere alla propria configurazione di bit sulle linee di indirizzo da A_3 a A_7. Una struttura simile potrebbe essere utilizzata per registri di ingresso ma apportando due modifiche. Invece che segnali $\overline{IOW}$ verrebbero usati segnali del tipo $\overline{IOR}$ (Input Output Read Not), e il circuito di uscita del registro dovrebbe includere un buffer a tre stati. Infatti tutti i dispositivi del microcomputer che possono inviare informazioni al bus dati devono avere un buffer a tre stati.

Buffer a tre stati

Il buffer a tre stati, come suggerisce il nome, ha tre stati logici. Due sono uguali a quel-

Figura 2.9. Registro con buffer d'uscita a tre stati

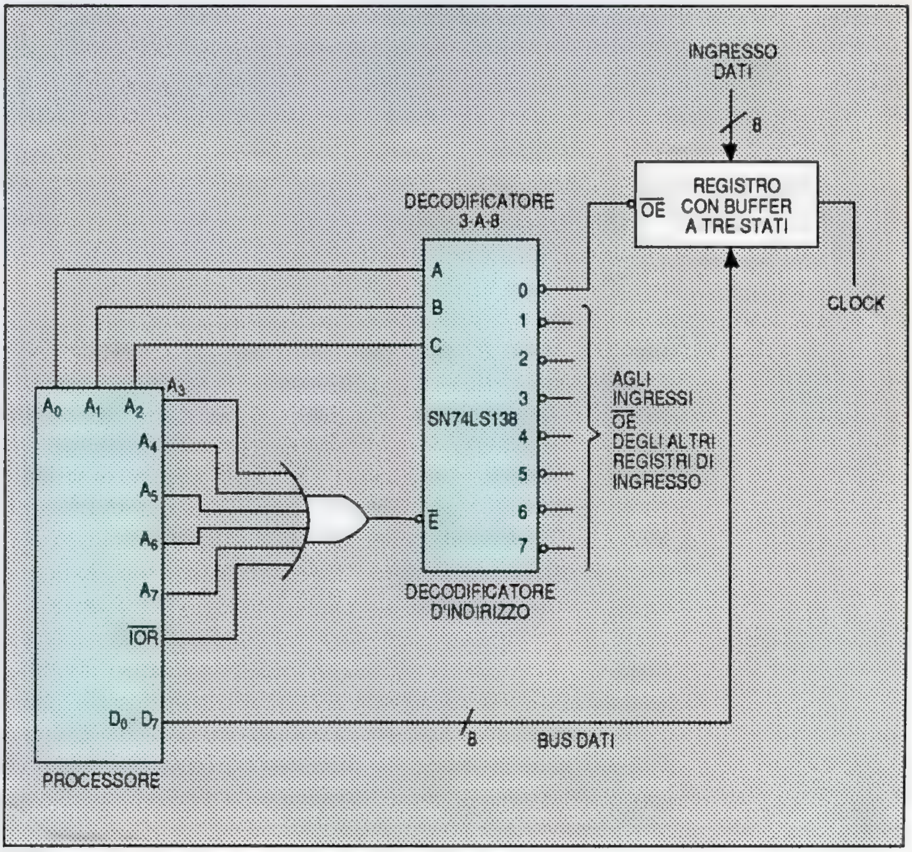

Solo un dispositivo può essere collegato col bus dati in un dato istante
REGISTRO FLIP-FLOP
BUFFER A TRE STRATI
INGRESSI DATI
D_3
D_2
D_1
D_0
D
Q
USCITA DATI VERSO IL BUS DATI
CLOCK
$\overline{OE}$

Il buffer a tre stati non ha solo i due stati logici 1 e 0 di una porta logica, ma possiede anche un terzo stato ad alta impedenza. Il terzo stato si comporta come un circuito aperto in modo tale da isolare il registro dal bus dati.

li di ogni altro dispositivo digitale: basso (0) o alto (1). Il terzo stato è uno stato di alta impedenza, ovvero essenzialmente un circuito aperto; cioè, in questo stato, il buffer sembra un interruttore aperto ai segnali sul suo ingresso. Questo permette al buffer di collegare o scollegare i segnali alla linea dati. Ciò è mostrato in *figura 2.9* unitamente ad un registro a 4 bit composto da 4 flip-flop. Come al solito il segnale di clock comune e le 4 linee dati in parallelo sono ingressi del registro. Ci sono 4 linee dati di uscita in parallelo che si collegano al bus dati, ma che saranno connesse con le uscite del registro solo se il buffer a tre stati è attivato. Il segnale che attiva il buffer in uscita viene indicato come $\overline{OE}$ (Output Enable Not). Il segnale $\overline{OE}$ deve essere 0 perché il buffer colleghi le uscite del registro con le linee dati. Se $\overline{OE}$ è ad 1, il buffer è nello stato di alta impedenza. In questo stato, i flip-flop non forniscono tensione sulle linee; perciò essi non ricevono né inviano corrente alle linee dati. Così, il buffer a tre stati isola il registro dalle linee dati.

Questa caratteristica è importante nei microcomputer perché molti componenti possono inviare informazioni sul bus dati, ma è importante che solo un dispositivo sia connesso col bus dati in un qualsiasi istante. Se due o più dispositivi potessero inviare sulle linee segnali di 1 e di 0 contemporaneamente, i circuiti di uscita dei dispositivi potrebbero essere danneggiati e i valori dei dati non sarebbero assolutamen-

Figura 2.10. Sistema d'ingresso di un computer


INGRESSO DATI
8
REGISTRO CON BUFFER A TRE STATI
OE
DECODIFICATORE 3-A-8
A
B
C
0
1
2
3
4
5
6
7
E
SN74LS138
AGLI INGRESSI OE DEGLI ALTRI REGISTRI DI INGRESSO
CLOCK
DECODIFICATORE D'INDIRIZZO
A0 A1 A2 A3
A4
A5
A6
A7
IOR
D0 - D7
8
BUS DATI
PROCESSORE

Il buffer a tre stati, basato su di un segnale di controllo, assicura che venga collegato al bus dati un solo dispositivo alla volta.

te affidabili. Il microprocessore può controllare quale dispositivo è connesso col bus dati inviando un segnale che attiva l'uscita al buffer a tre stati solo per il dispositivo prescelto. Perciò deve essere utilizzato uno schema di indirizzamento in modo che il microprocessore possa essere sicuro che è stato attivato il dispositivo esatto.

La *figura 2.10* illustra come questo può essere fatto per i registri di ingresso. Vengono utilizzati la stessa struttura del decodificatore e lo stesso schema di indirizzamento mostrati in *figura 2.8*, ma il decodificatore è attivato quando $\overline{IOR}$ è basso (invece che $\overline{IOW}$). Poiché solo un'uscita del decodificatore (come determinato dall'indirizzo) è bassa in un qualsiasi istante, solo il segnale $\overline{OE}$ per un registro di ingresso è a 0 in un qualsiasi istante e quindi solo un registro può inviare i suoi dati al processore.

Si noti che lo stesso indirizzo che viene utilizzato in *figura 2.10* per un registro di ingresso può essere utilizzato per un registro di uscita come in *figura 2.8*. Comunque entrambi i registri di ingresso e di uscita non saranno mai attivati contemporaneamente perché i segnali $\overline{IOW}$ e $\overline{IOR}$ controllano l'attivazione del decodificatore.

Memorie

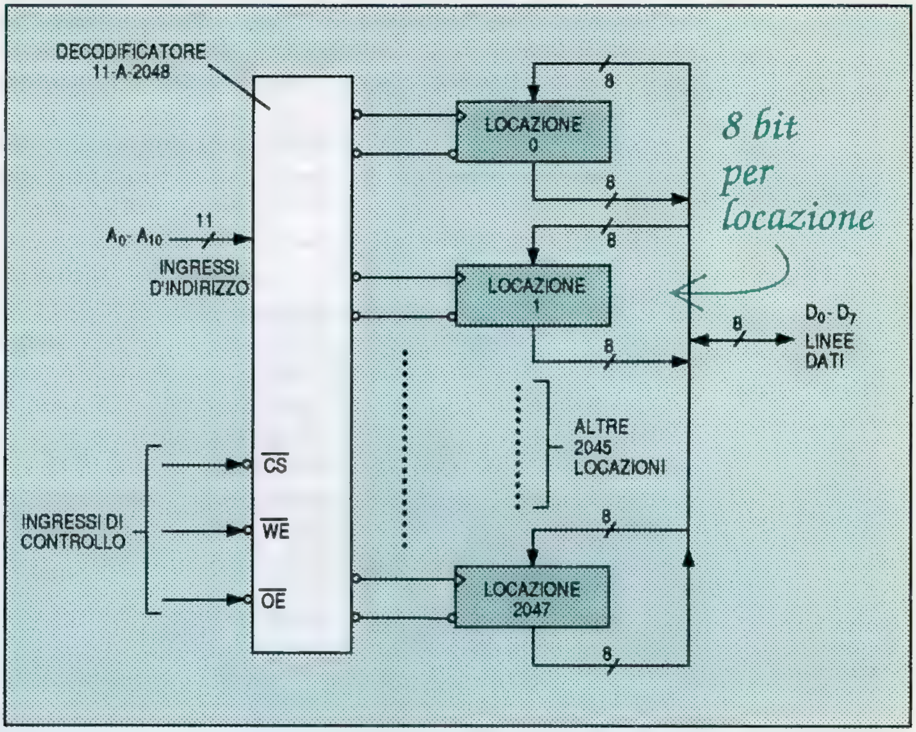
I registri sono solo una delle forme di memorizzazione presenti nella struttura di un computer. Si ha bisogno anche di dispositivi di memorizzazione che possano immagazzinare una grande quantità di codici ad n-bit (solitamente codici ad 8 bit) per contenere le istruzioni del programma, i dati e le costanti del sistema. Questi dispositivi vengono chiamati memorie. Generalmente essi possiedono un numero di locazioni che è una potenza del numero 2 ed hanno 1, 4 o 8 bit immagazzinati in ogni locazione. Spesso ci si riferisce alla quantità di celle di memoria in termini di K bit o K byte dove 1 K è uguale a 1024, cioè 2^{10}. I dispositivi di memoria sono disponibili in circuiti integrati monolitici da 1 K fino a 256 K bit.

Memoria di lettura-scrittura

La RAM viene utilizzata per memorizzazione a lungo termine di istruzioni dei programmi, dati e costanti. Essa consiste di una matrice di celle di memorizzazione per un dato numero di parole moltiplicato per un numero di bit per parola. È possibile sia scrivere in essa che leggere da essa. Un codice d'indirizzo applicato alla

In *figura 2.11* è mostrata l'organizzazione di una memoria a 2048 (2^{11}) locazioni (8 bit per locazione). Si tratta di una memoria di lettura-scrittura ad accesso casuale, solitamente chiamata più semplicemente memoria ad accesso casuale (RAM, da Random Access Memory). I segnali di controllo per il funzionamento della memoria includono l'abilitazione dell'uscita ($\overline{OE}$), l'abilitazione di scrittura ($\overline{WE}$) e la selezione del chip ($\overline{CS}$). Il decodificatore 11-a-2048 fa parte del dispositivo di memoria. Se $\overline{CS}$ non è a 0 il decodificatore non è attivato e quindi non si può né leggere né scrivere una locazione. Se $\overline{CS}$ è a 0 la memoria può essere letta o scritta a seconda dello stato di $\overline{WE}$ e $\overline{OE}$. Per leggere dalla memoria, $\overline{WE}$ deve essere ad 1 e $\overline{OE}$ deve essere a 0. Il contrario vale per un'operazione di scrittura: cioè $\overline{WE}$ deve essere 0 e $\overline{OE}$ deve essere 1. Alcuni circuiti di memoria non hanno l'ingresso $\overline{OE}$, così $\overline{CS}$ e $\overline{WE}$ controllano anche l'operazione di lettura. In questo caso, $\overline{WE}$ sta ad 1 (per la lettura) a meno che venga dato uno specifico comando di scrittura per portare $\overline{WE}$ a 0. L'altro ingresso al dispositivo di memoria è un codice di indirizzo a 11 bit per selezionare una qualsiasi delle 2048 posizioni. La memoria può essere pensata come un insieme formato da 2048 registri di ingresso come quelli precedentemente trattati, che possono essere collegati fra loro come registri di uscita. Il decodificatore 11-a-

Figura 2.11. Struttura di un circuito di memoria


DECODIFICATORE
11-A-2048
A_0-A_{10}
11
INGRESSI
D'INDIRIZZO
LOCAZIONE 0
LOCAZIONE 1
LOCAZIONE 2047
8
8 bit per locazione
D_0-D_7
LINEE
DATI
ALTRE
2045
LOCAZIONI
INGRESSI DI
CONTROLLO
$\overline{CS}$
$\overline{WE}$
$\overline{OE}$

memoria localizza la parola specifica.

2048 che genera il segnale di latch per l'abilitazione di ingresso o di uscita per ciascuno di questi registri interni (o locazioni di memoria) è semplicemente una versione molto ampliata del decodificatore 3-a-8 discusso nelle *figure 2.8* e *2.10*. Una volta decodificato l'indirizzo, la locazione di memoria è selezionata. A questo punto la locazione può ricevere dati (operazione di scrittura) dal processore (se $\overline{CS}$ è a 0 e $\overline{WE}$ è a 0) oppure può inviare dati (operazione di lettura) al processore (se $\overline{CS}$ è a 0 e $\overline{OE}$ è a 0). I flip-flop che formano ciascuna locazione di memoria generalmente non sono flip-flop del tipo con clock, ma sono più simili ai flip-flop di tipo RS trattati nel primo capitolo.

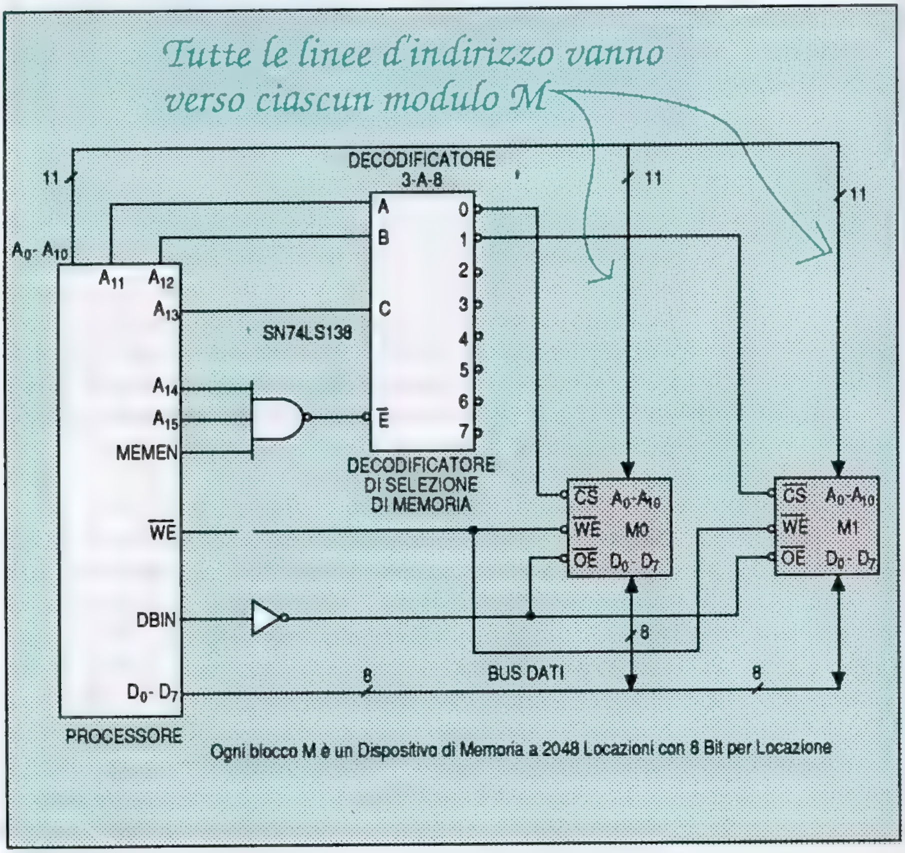
Indirizzamento della memoria

Poiché la memoria è essenzialmente una serie ampliata di registri di ingresso/uscita, essa viene collegata al processore utilizzando un metodo molto simile a quello delle strutture delle *figure 2.8* e *2.10*. Il collegamento al processore di due memorie del tipo mostrato in *figura 2.11* è illustrato in *figura 2.12*. Ci sono 8 posizioni disponibili, ma per semplicità ne vengono mostrate solo 2. Ogni blocco contrassegnato con una M è una memoria a 2048 locazioni con un decodificatore di indirizzo integrale 11-a-2048 come quello mostrato in *figura 2.11*. Gli 11 bit di indirizzo meno significativi vanno a ciascuna memoria per indicare quale delle 2048 locazioni è richiesta. I seguenti 3 bit di indirizzo (A_{11} - A_{13}) vanno agli ingressi di selezione del decodificatore 3-8 per determinare quale delle 8 possibili memorie è abilitata. I 2 bit di indirizzo più significativi (A_{14} - A_{15}) vanno ad una porta NAND insieme ad un segnale chiamato MEMEN (Memory Enable, abilitazione della memoria) per determinare quando il decodificatore 3-a-8 è attivato per decodificare un indirizzo al fine di generare un'uscita 0 su una delle sue linee. Questo è un metodo simile a quello mostrato nelle *figure 2.8* e *2.10*.

Il processore utilizza il segnale MEMEN per indicare ai circuiti esterni quando sta per verificarsi un'operazione di memoria. Fornisce anche un segnale $\overline{WE}$ e un segnale DBIN (Data Bus In) per controllare l'operazione di lettura e di scrittura della memoria. Poiché questo processore manda DBIN a 1 per un'operazione di lettura, viene inserito sulla linea un invertitore in modo tale che lo 0 richiesto appaia sull'ingresso $\overline{OE}$ della memoria. Con tutti i collegamenti di *figura 2.12* effettuati, una qualsiasi delle 16.384 locazioni mostrate (2048 x 8 = 16.384) può essere indirizzata individualmente e casualmente dal processore . La memoria M0 viene abilitata quando i 5 bit più significativi (A_{11} - A_{15}) dell'indirizzo sono 11000. La memoria M1 viene abilitata quando A_{11} - A_{15} sono 11001. Perciò la gamma di indirizzi per le 2048 locazioni in M0 va da $1100\ 0000\ 0000\ 0000_2$ ($C000_{16}$) a $1100\ 0111\ 1111\ 1111_2$ ($C7FF_{16}$). Analogamente M1 risponde ad indirizzi nella gamma da $1100\ 1000\ 0000\ 0000_2$ ($C800_{16}$) a $1100\ 1111\ 1111\ 1111_2$ ($CFFF_{16}$). Con l'aggiunta di altri circuiti di memoria controllati dalle altre 6 uscite del decodificatore 3-a-8 potranno essere aggiunte

Il processore (CPU) emette un segnale di controllo (MEMEN) per indicare ai circuiti esterni che si sta verificando un'operazione di memoria. I segnali WE e DBIN controllano se l'operazione è di scrittura o di lettura.

Figura 2.12. Sistema di memoria del computer


11
A_0- A_{10}
A_{11}
A_{12}
A_{13}
A_{14}
A_{15}
MEMEN
$\overline{WE}$
DBIN
D_0- D_7
PROCESSORE
DECODIFICATORE 3-A-8
A
B
C
$\overline{E}$
0
1
2
3
4
5
6
7
SN74LS138
DECODIFICATORE DI SELEZIONE DI MEMORIA
11
11
$\overline{CS}$
A_0-A_{10}
$\overline{WE}$
M0
$\overline{OE}$
D_0 - D_7
$\overline{CS}$
A_0-A_{10}
$\overline{WE}$
M1
$\overline{OE}$
D_0 - D_7
8
8
8
BUS DATI

Tutte le linee d'indirizzo vanno verso ciascun modulo M

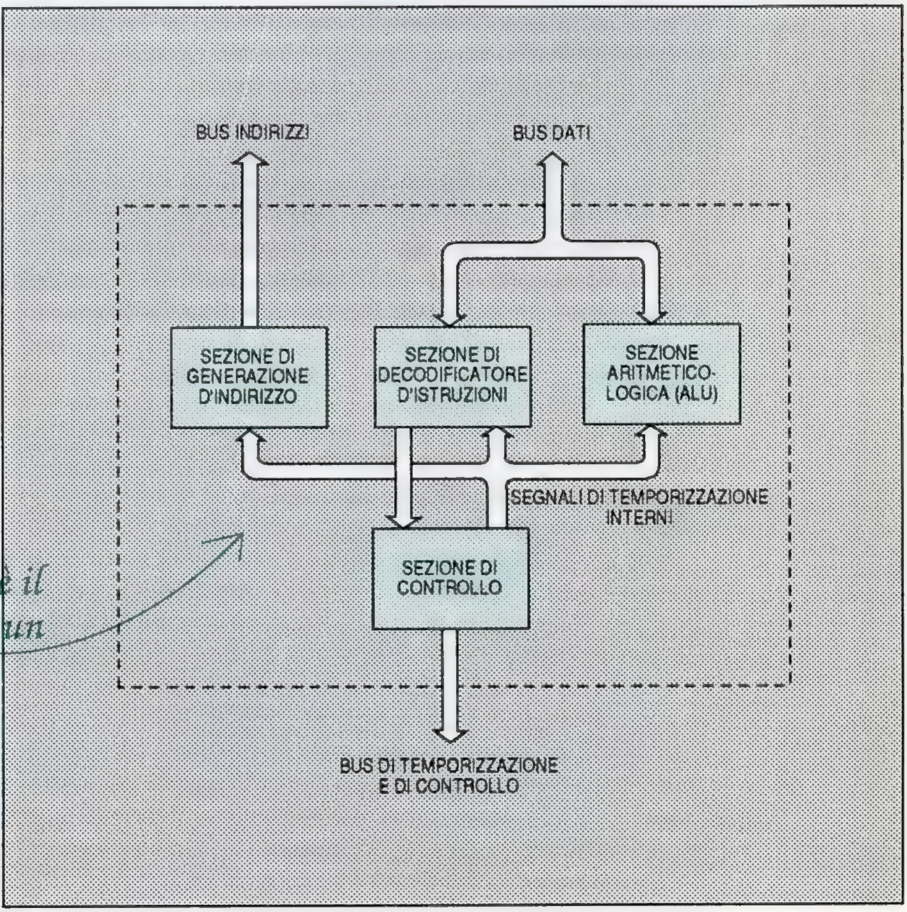
Ogni blocco M è un Dispositivo di Memoria a 2048 Locazioni con 8 Bit per Locazione

locazioni di memoria con indirizzi da $1101\ 0000\ 0000\ 0000_2$ ($D000_{16}$) a $1111\ 1111\ 1111\ 1111_2$ ($FFFF_{16}$).
Questo procedimento indica un metodo comunemente usato per incrementare la memoria di un microcomputer. Lo schema di indirizzamento è progettato per un numero massimo di memorie piuttosto elevato, ma poi vengono installati solo pochi circuiti integrati di memoria. Se più avanti si vorrà espandere la capacità di memoria, basterà inserire ulteriori IC (supponendo che siano già installati gli zoccoli appositi) .

Memoria a sola lettura

L'esempio di *figura 2.12* riguarda l'uso dei dispositivi di memoria che possono essere scritti o letti. Un altro tipo di memoria che si trova comunemente nei microcomputer è la memoria a sola lettura (ROM, da Read-Only Memory). Essa viene usata per memorizzare istruzioni di un programma e dati che rimangono costanti. Lo schema di indirizzamento e di lettura della memoria è lo stesso già descritto per la memo-

Figura 2.13. Schema a blocchi semplificato di un processore


BUS INDIRIZZI
BUS DATI
SEZIONE DI GENERAZIONE D'INDIRIZZO
SEZIONE DI DECODIFICATORE D'ISTRUZIONI
SEZIONE ARITMETICO-LOGICA (ALU)
SEGNALI DI TEMPORIZZAZIONE INTERNI
SEZIONE DI CONTROLLO
BUS DI TEMPORIZZAZIONE E DI CONTROLLO

La CPU è il "cuore" di un computer

La ROM (Memoria a sola lettura) viene indirizzata e letta come una RAM. Non si può scrivere su di essa. Le sue informazioni vengono memorizzate permanentemente quando essa viene costruita: perciò non è volatile.

Le PROM sono ROM che possono essere programmate anche dopo la costruzione; le EPROM sono ROM che possono essere riprogrammate dopo essere state cancellate.

ria di lettura-scrittura, eccetto per il fatto che in questo caso non c'è bisogno di un segnale che attivi la scrittura.

Il vantaggio di utilizzare una ROM è che essa non è volatile, mentre lo è la maggior parte delle memorie di lettura-scrittura. Una memoria volatile perde tutte le sue informazioni non appena viene interrotta l'alimentazione che la fa funzionare. Internamente la ROM è un circuito combinatorio a larga scala, formato dallo stesso decodificatore n-a-2^n sia per gli ingressi di indirizzo che per i buffer di uscita a tre stati controllati dai segnali $\overline{CS}$ ed $\overline{OE}$. Comunque, la "memoria" consiste di una configurazione di fili prefissata che pone in collegamento le uscite delle 2^n porte AND della logica combinatoria del decodificatore con le porte OR di uscita, la quale è pilotata dai buffer a tre stati. Per la ROM il cablaggio è fissato durante il processo di fabbricazione e non può essere alterato. La PROM (Programmable ROM) può essere programmata dopo la fabbricazione, ma solo per una volta; poi essa diviene come una ROM. La EPROM (Erasable PROM, PROM cancellabile) è programmata elettricamente per la memorizzazione permanente (sola lettura), ma le configurazioni di memoria possono essere cancellate con l'esposizione per un tempo sufficiente ai raggi ultravioletti. La memoria quindi può essere programmata di nuovo. Sia la PROM che la EPROM richiedono un dispositivo hardware speciale per la programmazione.

Processori

Il componente centrale di controllo nella struttura di un calcolatore è il processore. Un processore su singolo circuito integrato viene chiamato microprocessore. Le quattro parti generali di questo dispositivo molto complesso sono illustrate in *figura 2.13*. Ognuna di queste parti svolge una determinata funzione nei cicli di prelievo e di esecuzione delle istruzioni menzionati precedentemente.

Circuiteria di controllo

L'unità di controllo è un sistema digitale sequenziale che fornisce la funzione di controllo centrale per il sottosistema del processore. Essa svolge questa funzione tramite il fetch e l'esecuzione delle istruzioni, sotto la direzione del decodificatore d'istruzioni.

La sezione di controllo è costituita dall'unità centrale di controllo del processore. Il dispositivo di controllo è un sistema digitale sequenziale che svolge la funzione di prelievo ed esecuzione delle istruzioni. Esegue il ciclo di fetch (prelievo) fornendo l'indirizzo dell'istruzione da eseguire sul bus indirizzi e mandando segnali di controllo della lettura della memoria sul bus di controllo e di temporizzazione. Quindi dispone il bus dati del processore nel modo operativo di ingresso e memorizza l'istruzione proveniente dalla memoria nel registro istruzioni. Tutto questo descrive completamente il trasferimento di un'istruzione semplice. Alcune istruzioni, invece, possono richiedere la lettura di 2 o 3 locazioni di memoria. La funzione di controllo tiene conto di queste richieste e continua a leggere la memoria in locazioni successive finché l'istruzione completa è nel registro istruzioni. Il decodificatore di istruzioni informa il dispositivo di controllo su che istruzione sta per essere eseguita. Se l'istruzione richiede la lettura o la scrittura di una memoria o un'operazione di ingresso/uscita, il dispositivo di controllo fornisce l'indirizzo della memoria o dell'ingresso/uscita e i segnali di controllo appropriati per leggere o scrivere l'informazione richiesta dall'istruzione. Se l'istruzione comporta un'operazione aritmetica o logica il dispositivo di controllo attiva l'apposita sezione dell'unità aritmetico-logica (ALU).

Queste sequenze operative sono incorporate nel progetto della logica sequenziale che costituisce la sezione di controllo.

Sezione di decodifica delle istruzioni

La sezione di decodifica delle istruzioni possiede un registro istruzioni per la memorizzazione temporanea del codice istruzione ottenuto dalla memoria. Essa contiene anche un decodificatore logico combinatorio che interpreta questo codice per la sezione di controllo.

Sezione di generazione degli indirizzi

Il contatore di programma (program counter) è uno speciale registro che mantiene l'indirizzo dell'istruzione presente o successiva del programma a seconda del punto in cui si trova il computer nel suo ciclo di funzionamento.

La sezione che genera gli indirizzi ha uno o più registri per tener nota delle locazioni dei dati necessari per l'esecuzione delle istruzioni e uno speciale registro chiamato contatore di programma che contiene l'indirizzo della prossima istruzione che deve essere trasferita. Poiché viene letta una locazione di memoria durante il fetch dell'istruzione, ogni volta viene aggiunto un 1 al contenuto del contatore di programma in modo che esso contenga l'indirizzo della locazione di memoria immediatamente successiva nella sequenza del programma. Il contatore di programma è un tipo speciale di registro chiamato contatore. Esso non solo può contenere il codice di indirizzo come un normale registro, ma può anche sommare un 1 al valore memorizzato (può incrementare il valore). In altre parole, esso conta. Questo è tutto ciò di cui si ha bisogno per tenere conto della locazione dell'istruzione, poiché le istruzioni normalmente si susseguono una dopo l'altra nella parte di memoria utilizzata per le istruzioni del programma. Talvolta le istruzioni trasferite richiedono che il successivo fetch di istruzione avvenga da una locazione di memoria fuori da questa sequenza. Se è così, l'istruzione conterrà l'indirizzo della locazione fuori sequenza ed esso verrà caricato nel contatore del programma. Il prelievo dell'istruzione avverrà quindi alla nuova locazione del contatore di programma, il quale continuerà a prelevare le istruzioni successive in sequenza finché gli verrà comandato di ritornare alla sequenza originale o di iniziare una nuova sequenza.

Unità aritmetico-logica

Una ALU contiene circuiti come gli addizionatori e le porte logiche per svolgere le operazioni aritmetiche e logiche richieste dal processore.

La ALU è qualcosa di più complesso del decodificatore di istruzioni e del generatore di indirizzi. Essa contiene sommatori, sottrattori, porte logiche ed altri dispositivi per svolgere operazioni aritmetiche quali addizione, sottrazione, valore assoluto, moltiplicazione e operazioni logiche quali OR, NOT, AND e così via. La sezione ALU fornisce anche una serie di registri per la memorizzazione di dati all'interno del processore, perché a questi dati si possa accedere e fare riferimento velocemente.
Solitamente l'insieme dei circuiti interni al processore è di scarso interesse per i progettisti e gli utilizzatori dei sistemi a microcomputer, in quanto il processore è semplicemente un componente. Purché un progettista sappia come collegare il processore in un sistema con gli altri componenti (come in *figura 2.8*, in *figura 2.10* e in *figura 2.12*), ciò che succede all'interno del processore è di scarso interesse. Anche i programmatori devono solo sapere quali locazioni di memoria interna sono disponibili per le operazioni del programma per poter scrivere con risultati positivi programmi per un dato sistema.
Ci sono molti componenti digitali che vengono usati nel computer ed in altri siste-

mi digitali che non sono stati trattati in questi due primi capitoli. Comunque, i dispositivi più comuni sono stati introdotti, perciò è ora possibile considerare le tecniche per trovare i guasti nei sistemi digitali. Esse verranno presentate nel capitolo seguente.

CHE COSA ABBIAMO IMPARATO?

1. Sono stati studiati i tipi fondamentali di sistemi digitali. Essi consistono di sistemi logici combinatori (senza memoria) e sistemi logici sequenziali (con memoria).
2. Nei sistemi più complessi vengono utilizzati sia i circuiti combinatori che quelli sequenziali.
3. La struttura di sistema che sta diventando sempre più comune è la struttura del computer.
4. Le funzioni fondamentali di tutti i computer sono la memoria, l'ingresso, l'uscita e il processore.
5. I registri con ingresso e uscita in parallelo sono costituiti da flip-flop con un clock comune.
6. Un buffer a tre stati agisce come un interruttore per collegare o scollegare i segnali dal bus.
7. Una ROM memorizza un'informazione permanentemente in modo che essa non vada perduta qualora venga staccata l'alimentazione.
8. Una RAM invece è un dispositivo in cui si può leggere e scrivere e viene utilizzata per immagazzinare informazioni che possono variare.
9. Ogni locazione nelle memorie RAM e ROM ha un indirizzo diverso cosicché può essere raggiunta dal processore in modo casuale o singolarmente.
10. Il compito principale del processore è quello di prelevare (to fetch) e di eseguire le istruzioni.

Quiz del Capitolo 2

1. Lo scopo di una funzione di elaborazione in un sistema è:
   a. Immagazzinare un'informazione di cui si ha bisogno successivamente.
   b. Generare uscite in base agli ingressi passati e presenti.
   c. Portare un'informazione all'interno del sistema, agire su questa, e farla uscire dal sistema stesso.

2. La memoria è necessaria in un sistema se:
   a. Le uscite sono una funzione degli ingressi.
   b. Il sistema deve memorizzare costanti o dati del sistema.
   c. Le uscite sono una funzione degli ingressi passati.
   d. Le uscite sono una funzione degli ingressi passati o eventi che sono già stati implementati.
   e. Tutte le risposte precedenti.
   f. Le risposte b, c, d.

3. Il processore in un computer svolge:
   a. La funzione di memoria principale del sistema.
   b. La funzione di trasferire dati solo dalla memoria al processore o viceversa.
   c. La funzione di svolgere solo operazioni aritmetiche o logiche.
   d. La funzione di elemento di controllo principale del sistema.

4. Lo scopo del ciclo di prelievo (fetch) in un computer è quello di:
   a. Ottenere dati in ingresso.
   b. Ottenere dati in memoria.
   c. Ottenere un'istruzione.
   d. Implementare un'operazione specifica.

5. Quando un processore è nel suo ciclo esecutivo:
   a. Non vengono svolte operazioni di lettura in memoria.
   b. Il codice di istruzione è nel registro istruzioni.
   c. Vengono svolte operazioni solo dalla sezione ALU.
   d. Il contatore di programma non è coinvolto nel ciclo.

6. Un circuito di memoria che ha 9 ingressi indirizzabili quante locazioni di memoria ha?
   a. 256
   b. 512
   c. 1024
   d. Non determinabile.

7. Se in *figura 2.12* fosse stata utilizzata una porta OR invece che una porta NAND e fosse stato inserito un invertitore sulla linea MEMEN, quale sarebbe stato il primo indirizzo in notazione esadecimale della prima locazione del circuito di memoria M0?
   a. 0
   b. 1000
   c. C000
   d. F000

8. Nella situazione dell'esercizio precedente, quale sarebbe stato l'ultimo indirizzo in notazione esadecimale dell'ultima locazione del circuito di memoria M1?
   a. 0FFF
   b. 3FFF
   c. CFFF
   d. FFFF

9. Se un circuito di memoria fornisce 8192 locazioni con 1 bit per locazione, quanti di questi circuiti saranno necessari per avere una memoria con 16384 locazioni e 8 bit per locazione?
   a. 2
   b. 8
   c. 16
   d. 32

10. Se in *figura 2.12* si usasse un processore che non fornisce il segnale MEMEN ma il segnale $\overline{\text{MEMR}}$ (0 per un'operazione di lettura dalla memoria) e $\overline{\text{MEMW}}$ (0 per un'operazione di scrittura della memoria), quale porta logica si potrebbe usare con ingressi $\overline{\text{MEMR}}$ e $\overline{\text{MEMW}}$ per ottenere MEMEN?
    a. OR
    b. NOR
    c. AND
    d. NAND
    e. OR Esclusivo.

FONDAMENTI DI RICERCA DEI GUASTI

SCOPO DI QUESTO CAPITOLO

Negli ultimi due capitoli sono stati passati in rassegna i principi fondamentali di funzionamento dei circuiti elettrici, dei componenti digitali e dei sistemi digitali. Con questa conoscenza di base è possibile cominciare ad esaminare le tecniche fondamentali per la ricerca dei guasti nei sistemi digitali e la loro riparazione.

Il metodo d'approccio alla eliminazione di guasti nei sistemi elettronici è molto simile a quello utilizzato da un investigatore. Esso consiste nella raccolta delle prove (svolgendo delle misurazioni) e nel ragionamento sulla base di queste per individuare con precisione nel sistema il componente, il filo o la connessione colpevole. Fortunatamente, questo è un procedimento che può essere imparato facilmente.

Anche in sistemi molto complicati i guasti sono quasi sempre causati da qualche semplice difetto che può essere trovato con un'attrezzatura di prova relativamente economica e semplice. I guasti fondamentali che verranno considerati in questo capitolo sono il corto circuito, il circuito aperto, l'errato livello di tensione e il guasto di un componente.

CORTO CIRCUITO

Un corto circuito, detto anche semplicemente "corto", è un collegamento elettrico che non dovrebbe esistere nel sistema. Produce lo stesso effetto di un filo deliberatamente collegato tra due punti.

Corto in un circuito in corrente continua

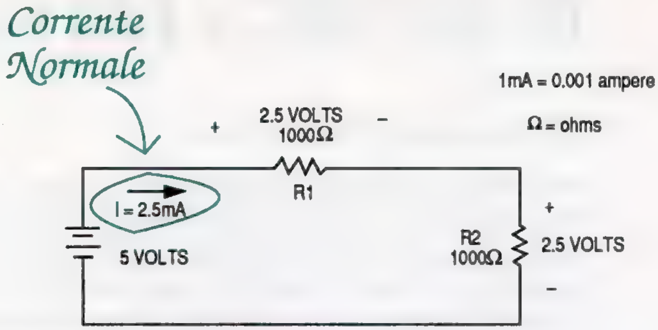
Un corto circuito in un circuito è semplicemente un collegamento elettrico che non era previsto. Esso fa sì che la corrente non circoli lungo il cammino assegnatogli: cioè alcuni componenti non ricevono corrente mentre è possibile che altri vengano sovraccaricati.

L'effetto di un corto su un circuito in corrente continua si può vedere esaminando il semplice circuito di *figura 3.1*. Il circuito in *figura 3.1a* è simile a quello utilizzato per introdurre i concetti di tensione, corrente, resistenza e potenza nel capitolo 1.

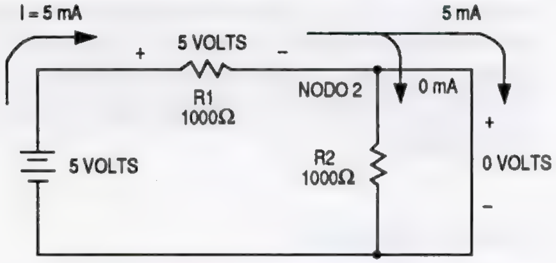
Poiché le due resistenze sono collegate in serie con un alimentatore, attaverso ciascuna di esse passerà la stessa corrente. Inoltre, poiché sono di eguale valore (1000 ohm ciascuna), si avrà la stessa caduta di tensione ai capi di ognuna. In *figura 3.1a* la tensione di 5 volt proveniente dall'alimentatore è applicata ad un circuito la cui resistenza totale è di 2000 ohm, cosicché attraverso le resistenze passa una corrente di 0,0025 ampere o 2,5 milliampere. Questo fa sì che la caduta di tensione su ogni resistenza sia di 2,5 volt.

In *figura 3.1b* il circuito è cambiato leggermente mettendo in corto circuito R2. Poiché un corto è come un collegamento di fili e dato che un filo offre pochissima resistenza al passaggio di corrente, tutta la corrente proveniente dall'alimentatore passa attraverso R1 e il corto. Attraverso R2 non passa corrente cosicché la tensione ai capi di R2 è di 0 volt. Inoltre, poiché ora i 5 volt dell'alimentazione sono totalmen-

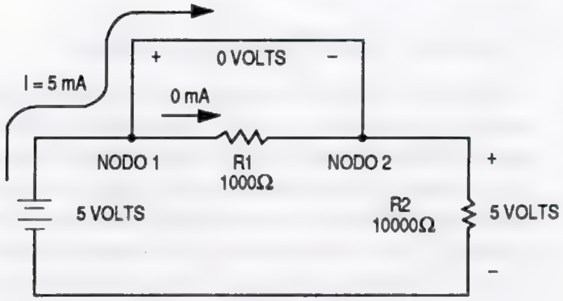
Figura 3.1. Effetto di un corto circuito in un semplice circuito in CC


Corrente Normale
1mA = 0.001 ampere
Ω = ohms
+
2.5 VOLTS
1000Ω
–
I = 2.5mA
R1
5 VOLTS
R2
1000Ω
+
2.5 VOLTS
–

a. Circuito Normale


I = 5 mA
5 mA
+
5 VOLTS
–
R1
1000Ω
NODO 2
0 mA
+
5 VOLTS
R2
1000Ω
0 VOLTS
–

b. Resistore Cortocircuitato R2


+
0 VOLTS
–
I = 5 mA
0 mA
NODO 1
R1
1000Ω
NODO 2
+
5 VOLTS
R2
10000Ω
5 VOLTS
–

c. Resistore Cortocircuitato R1

te applicati ai capi della sola R1, la corrente che passa attraverso R1 e il corto è di 5 milliampere. Questo è un altro fattore da considerare nella riparazione di un circuito, cioè l'eccessiva corrente attraverso R1 dovuta al corto potrebbe danneggiare R1. Perciò si dovrebbe misurare il valore di R1 per verificare che sia corretto.

Un corto circuito in un circuito in corrente continua può essere individuato misurando i cambiamenti delle tensioni in un nodo, delle correnti nel circuito o in un ramo, o della resistenza totale del circuito.

Si noti che esistono tre modi generali per scoprire che è avvenuto un corto. Per prima cosa la tensione ai capi di R2 è passata da 2,5 volt a 0 volt. Perciò misurando la tensione del nodo 2 rispetto alla massa, questa indicherà un corto fra quel nodo e la massa. (Si ricordi dal primo capitolo che un nodo di un circuito è il punto di connessione di due o più rami). La seconda indicazione della presenza di un corto è che la corrente proveniente dall'alimentatore è raddoppiata, la qual cosa indica che la resistenza del circuito è stata dimezzata il che, a sua volta, indica che una delle due resistenze è stata messa in corto e quindi esclusa dal circuito. L'indicazione della corrente però non è un metodo efficace come l'indicazione della tensione perché non identifica precisamente quale resistenza è stata messa in corto. Il terzo modo per scoprire il corto è quello di staccare l'alimentatore e misurare la resistenza tra il nodo 2 e la massa. Se la resistenza è di 0 ohm invece che di 1000 ohm deve esserci un corto tra il nodo 2 e la massa. Comunque, questa misurazione non è così semplice come la misurazione della tensione poiché essa richiede che venga sconnessa una parte del circuito, l'alimentatore, prima che possa venire eseguita la misurazione.

Un altro esempio di corto nel circuito di *figura 3.1a* è mostrato in *figura 3.1c*. In questo caso c'è un corto tra il nodo 1 e il nodo 2 ai capi di R1. Per individuare ed isolare il corto possono essere utilizzate le stesse misurazioni descritte precedentemente. Sebbene in *figura 3.1b* e in *figura 3.1c* il corto sia indicato come un corto esterno, gli effetti e le misurazioni sarebbero le stesse se la resistenza fosse andata in corto internamente.

Un'altra possibilità di corto in questo tipo di circuito è il percorso che va dal nodo 1 a massa attraverso l'alimentatore. Comunque, le conseguenze di questo corto lo isolerebbero perché o il corto si aprirebbe o l'alimentatore resterebbe danneggiato. In entrambi i casi l'alimentatore emetterebbe un elevato e momentaneo impulso di corrente con una tensione di uscita molto bassa.

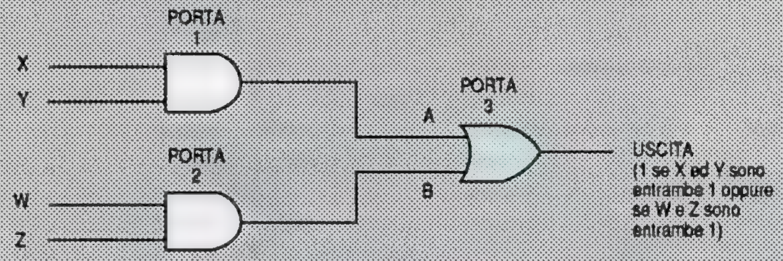
Corto in un circuito logico

Un corto circuito in un circuito logico digitale si verifica come uno stato logico permanente applicato ad una o più porte indipendentemente da ogni altro ingresso logico.

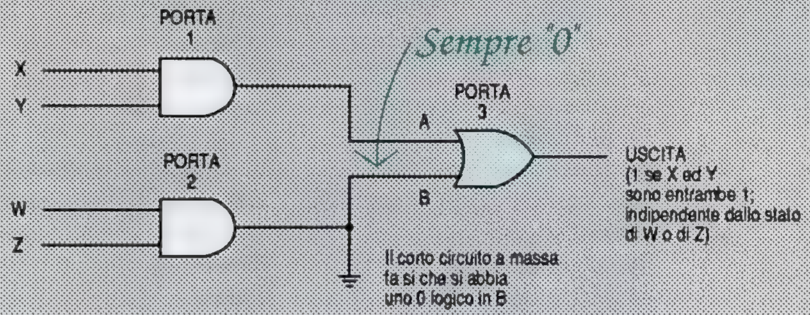
Un semplice circuito logico digitale senza un corto è mostrato in *figura 3.2a*. In *figura 3.2b* è mostrato un corto a massa che avviene da qualche parte sulla linea dall'uscita della porta 2 all'ingresso B della porta 3. L'effetto del corto è quello di produrre uno 0 logico permanente all'ingresso B della porta 3 (In queste discussioni il livello di massa o la tensione nulla sono uno 0 logico e la tensione di +5 volt è un 1 logico.) Perciò l'uscita della porta 3 e del circuito dipendono solo dalle condizioni dei segnali X e Y e non da quelle dei segnali W e Z. Nel circuito in condizioni normali (*figura 3.2a*), l'uscita del circuito sarà un 1 logico se X e Y sono entrambi 1 oppure se W e Z sono entrambi 1. Nel circuito guasto l'uscita non dipende dai livelli di W e Z e diventa 1 solo se X e Y sono entrambi 1.

È possibile localizzare la posizione

Osservando che le condizioni di W e Z non hanno effetti sull'uscita del sistema, si potrebbe dedurre che sulla linea d'uscita della porta 2 ci sia il valore 0. Se l'uscita della porta 2 fosse permanentemente fissa al livello 1, anche l'uscita del circuito do-


PORTA
1
X
Y
PORTA
2
W
Z
PORTA
3
A
B
USCITA
(1 se X ed Y sono entrambe 1 oppure se W e Z sono entrambe 1)

a. Circuito Logico Normale


Sempre "0"
PORTA
1
X
Y
PORTA
2
W
Z
PORTA
3
A
B
USCITA
(1 se X ed Y sono entrambe 1; indipendante dallo stato di W o di Z)
Il corto circuito a massa fa sì che si abbia uno 0 logico in B

b. Circuito Logico con un Corto Circuito

Figura 3.2. Effetto di un corto circuito in un circuito logico

generica di un corto circuito in un circuito logico tramite il ragionamento. Le misurazioni reali isoleranno completamente il problema.

vrebbe essere permanentemente fissa al livello 1, ma questo non si nota. Se uno degli ingressi alla porta 2 fosse fissato a 1, l'altro ingresso avrebbe ancora effetto sull'uscita; l'uscita del circuito diverrebbe 1 qualora andasse ad 1 l'ingresso attivo, ma anche questo non avviene. L'unico modo in cui l'uscita del circuito può essere indipendente sia da W che da Z e non essere ad ogni istante 1 è il verificarsi di un corto a massa sulla linea di ingresso di W, sulla linea di ingresso di Z o sulla linea tra l'uscita della porta 2 e l'ingresso B della porta 3. Basteranno solo 3 misure di tensione per determinare quale delle 3 linee ha subito il corto a massa. Anche senza fare ragionamenti, sette misurazioni di tensione (all'ingresso e all'uscita di ogni porta) bastano a determinare l'esistenza di un qualsiasi corto nel circuito. Una volta scoperti questi corti il riparatore deve compiere l'ulteriore passo di determinare la causa reale del corto; egli deve cioè scoprire se è difettosa la porta 2, la porta 3 o se il corto è avvenuto nei cablaggi. Si parlerà di questo un po' più avanti; ora verrà invece trattata la situazione opposta al corto, cioè la condizione di circuito aperto.

CIRCUITO APERTO

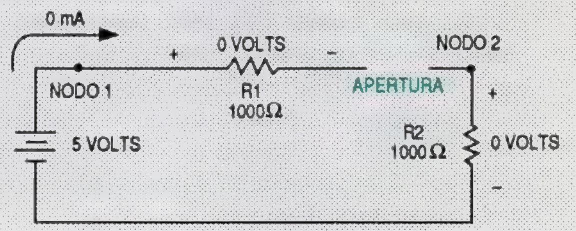
Il circuito aperto è un'interruzione nel cablaggio o in un componente. Il principale indizio di un'apertura nel circuito è che la corrente non può passare. Gli effetti che un'apertura provoca sui circuiti di *figura 3.1* e di *figura 3.2* sono mostrati in *figura 3.3*.

Apertura in un circuito in corrente continua

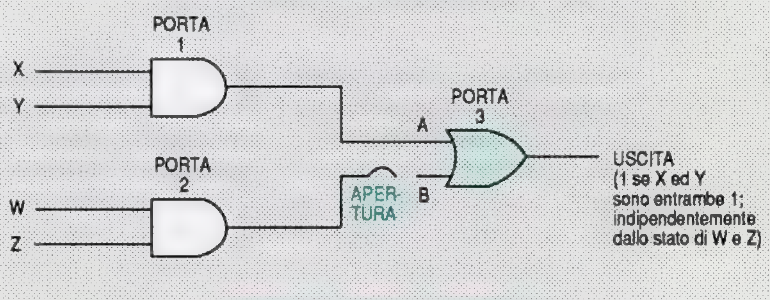
Un circuito aperto è un'apertura fisica nel cablaggio di un circuito o in uno dei suoi componenti. Esso viene indicato da una resistenza molto elevata nel circuito. La corrente inoltre non può circolare in un circuito in cui si è verificata un'apertura.

In *figura 3.3a* l'apertura si trova tra le due resistenze, cosicché il passaggio di corrente dall'alimentatore attraverso le due resistenze è 0; perciò la tensione ai capi delle due resistenze è 0. Così, se si pone nel circuito un misuratore di corrente, questo indicherà 0 milliampere. Ponendo un voltmetro tra il nodo 2 e la massa questo indicherà 0 volt (come nel caso di *figura 3.1b*). Se il voltmetro viene messo all'estremità destra di R1, prima dell'apertura, esso indicherà 5 volt (cosa che non sarebbe avvenuta nel caso del circuito di *figura 3.1b*). Se si effettua un controllo della resistenza tra il nodo 1 e la massa con l'alimentatore staccato lo strumento misuratore indicherà infinito, che è la resistenza di un circuito aperto. Ponendo invece l'ohmetro tra il nodo 2 e massa, questo indicherà 1000 ohm (invece di 0 ohm come nel caso di *figura 3.1b*). Perciò è possibile determinare se il guasto è causato da un'apertura o da un corto eseguendo il numero necessario di misurazioni. Le caratteristiche fondamentali dell'apertura sono il passaggio di corrente nullo nel circuito, l'insolita lettura sull'ohmetro (in alcuni casi infinito) e le insolite letture di valori della tensione in punti differenti del circuito.

Figura 3.3. Effetto di un'apertura in un circuito



a. Effetto di un' Apertura in un Semplice Circuito in CC



b. Effetto di un' Apertura in un Circuito Logico

Apertura in un circuito logico

Se il circuito aperto si verifica in un circuito digitale del tipo mostrato in *figura 3.2* il circuito non funzionerà in modo appropriato. Un esempio di guasto è mostrato in *figura 3.3b*. In questo esempio, l'apertura si trova sul filo che va dall'uscita della porta 2 all'ingresso B della porta 3; perciò la porta 2 non ha alcun controllo sulla porta 3. Cioè l'uscita del circuito è una funzione soltanto di X e Y e sarà 1 solo quando X e Y sono entrambi 1. Anche in questo caso l'uscita del circuito è completamente indipendente dai segnali su W e Z.

Per isolare un guasto in un circuito logico devono essere eseguite delle misurazioni perché, in molti casi, un'apertura in un circuito logico può dare lo stesso tipo di indicazioni di un corto circuito.

A questo punto è ovvio che il corto di *figura 3.2b* e l'apertura di *figura 3.3b* producono lo stesso effetto sul comportamento generale del sistema. L'unico modo per determinare se il guasto è un'apertura o un corto è quello di eseguire misurazioni della tensione sulla parte del circuito che comprende la porta 2. Un voltmetro posto sull'uscita della porta 2 in *figura 3.3b* indicherà 1 quando sia W che Z sono 1, il che vuol dire che la porta 2 sta funzionando in modo appropriato. Invece in *figura 3.2* l'uscita della porta 2 resterebbe fissa a 0 indipendentemente dai segnali su W e Z. Questo porta alla conclusione che, per la *figura 3.3b*, c'è un'apertura tra l'uscita della porta 2 e l'ingresso B della porta 3 o che la porta 3 è difettosa. L'apertura potrebbe essere isolata se la resistenza del ramo tra l'uscita della porta 2 e l'ingresso B della porta 3 fosse misurata con un ohmetro. Invece che indicare, come normalmente succede per un filo, 0 ohm, l'ohmetro indicherebbe una resistenza molto elevata o infinita, la qual cosa significherebbe che esiste un'interruzione nei collegamenti elettrici tra i due punti.

I casi di *figura 3.2* e di *figura 3.3* indicano che il semplice ragionamento basato sul funzionamento di un circuito non è sufficiente per isolare il guasto. Sapere che il circuito risponde in modo adeguato ai segnali X e Y e non ai segnali W e Z restringe il problema ad una zona più limitata del circuito ma non è possibile senza eseguire misurazioni determinare se è danneggiata la porta 2, se in parte lo è anche la porta 3 o se c'è un corto o un'apertura sulle linee del circuito. Perciò il ragionamento basato sul funzionamento del sistema indica la parte del circuito dove devono essere eseguite le misurazioni per riuscire finalmente ad isolare il guasto di un particolare componente o di un particolare filo.

LIVELLI DI TENSIONE NON CORRETTI

Tra la condizione di corto circuito e quella di circuito aperto ne esiste un'altra in cui ad un dato punto il segnale non è fisso né a 0 né ad 1, ma presenta un livello di tensione troppo basso per indicare con affidabilità 1, ed anche troppo alto per indicare con affidabilità 0. Naturalmente un sistema con un difetto di questo tipo non funzionerà in maniera corretta, ma, quel che è peggio, non funzionerà sempre allo stesso modo. Al fine di capire come può verificarsi un livello di tensione errato occorre svolgere un esame più approfondito di come lavorano i transistori nei circuiti digitali.

Circuito invertitore a transistore

La *figura 3.4a* mostra un circuito a transistore molto semplice e senza guasti. Que-

Figura 3.4. Guasti nei circuiti a transistori

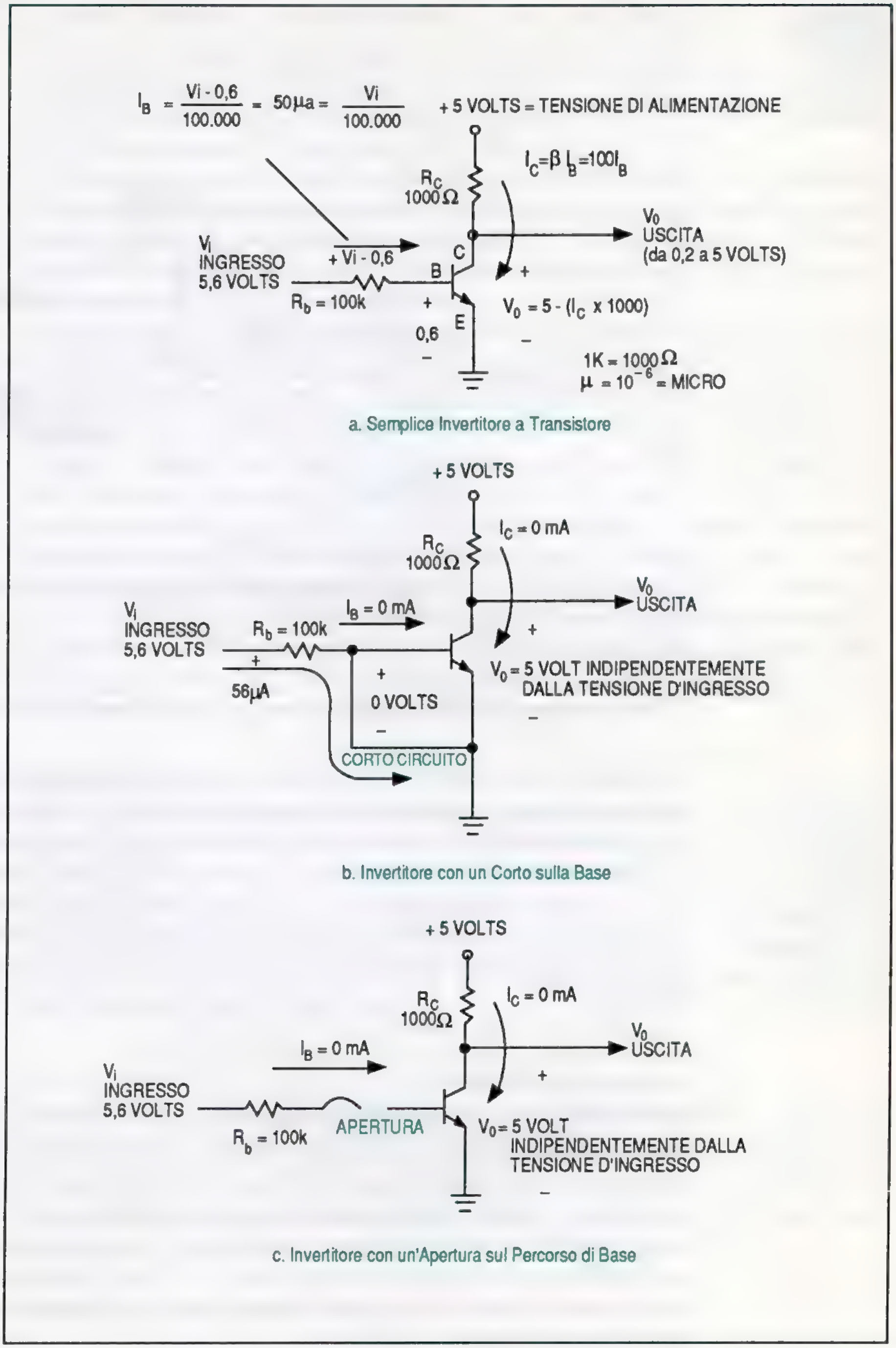
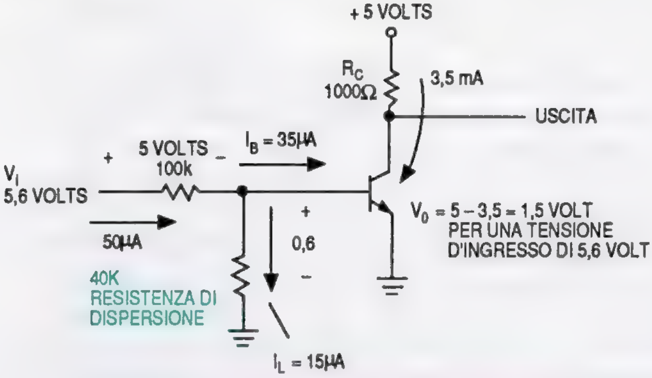

$I_B = \frac{Vi - 0,6}{100.000} = 50\mu a = \frac{Vi}{100.000}$
+ 5 VOLTS = TENSIONE DI ALIMENTAZIONE
$I_C = \beta I_B = 100 I_B$
R_C 1000 Ω
V_0 USCITA (da 0,2 a 5 VOLTS)
V_i INGRESSO 5,6 VOLTS
+ Vi - 0,6
C
B
E
R_b = 100k
+
0,6
–
$V_0 = 5 - (I_C \times 1000)$
1K = 1000 Ω
$\mu = 10^{-6}$ = MICRO
a. Semplice Invertitore a Transistore
+ 5 VOLTS
I_C = 0 mA
R_C 1000 Ω
V_0 USCITA
V_i INGRESSO 5,6 VOLTS
R_b = 100k
I_B = 0 mA
56µA
0 VOLTS
V_0 = 5 VOLT INDIPENDENTEMENTE DALLA TENSIONE D'INGRESSO
CORTO CIRCUITO
b. Invertitore con un Corto sulla Base
+ 5 VOLTS
I_C = 0 mA
R_C 1000 Ω
V_0 USCITA
I_B = 0 mA
V_i INGRESSO 5,6 VOLTS
R_b = 100k
APERTURA
V_0 = 5 VOLT INDIPENDENTEMENTE DALLA TENSIONE D'INGRESSO
c. Invertitore con un'Apertura sul Percorso di Base

Figura 3.4. Guasti nei circuiti a transistori
(continua)



d. Effetto di una Resistenza di Dispersione tra la Base e l'Emettitore

Utilizzando transistori con guadagno di corrente, i circuiti funzionano ad uno dei due livelli logici. Un semplice invertitore ne è un esempio.

sto circuito verrà utilizzato per illustrare il funzionamento del principale circuito digitale a transistori. La caratteristica fondamentale del funzionamento di un transistore è che esso produce un notevole guadagno di corrente. Se si applica un segnale all'ingresso in modo che venga prodotta una corrente di base la corrente del collettore sarà 50 o più volte maggiore della corrente di base. La tensione di uscita si ottiene moltiplicando la corrente del collettore (in ampere) per il valore della resistenza del collettore (in ohm) e sottraendo il valore risultante dalla tensione dell'alimentazione.

Funzionamento normale

Nell'esempio di *figura 3.4a* viene prodotta una corrente di base di 50 microampere (0,000050 ampere) dal valore logico 1 applicato al terminale di ingresso del circuito. Supponendo che il guadagno di corrente sia 100, questo produce una corrente di collettore di 5 milliampere che a sua volta causa una caduta di 5 volt ai capi della resistenza da 1000 ohm del collettore. Quando questi 5 volt vengono sottratti dai 5 volt dell'alimentazione, la tensione di uscita è di 0 volt, che è nella gamma dello 0 logico. Dunque, il circuito a transistore funziona come un invertitore perché si ha uno 0 in uscita quando c'è un 1 in ingresso.

D'altra parte, quando viene applicato all'ingresso uno 0 logico (segnale di 0 volt) la corrente di base è 0; perciò la corrente del collettore è 0. Ciò provoca una caduta di 0 volt attraverso la resistenza da 1000 ohm del collettore cosicché la tensione di uscita è 5 - 0 = 5 volt, valore di tensione che è compreso nella gamma dell' 1 logico. Anche in questo caso il transistore agisce come un invertitore cosicché uno 0 logico in ingresso produce un 1 logico in uscita. Circuiti più complicati possono essere trattati più o meno nello stesso modo per determinare gli effetti di un livello di ingresso 0 o 1 sulla tensione di uscita e sul livello logico.

La tensione tra la base e l'emettitore per un transistore di silicio saturato è ad un valore tra 0,6 volt e 0,7 volt, mentre tra il collettore e l'emettitore è a un valore compreso tra 0,2 volt e 0,4 volt.

C'e' un altro aspetto del funzionamento del transistore che non è stato considerato ancora: quando la corrente passa nella base del transistore la caduta di tensione interna tra la base e l'emettitore è compresa tra 0,6 e 0,7 volt se il transistore è al silicio. Perciò la tensione in ingresso deve essere più grande di questa tensione per provocare il passaggio della corrente di base.

Analogamente, quando la corrente del collettore sta scorrendo, la caduta di tensione interna tra il terminale del collettore e il terminale dell'emettitore impedisce di avere un'uscita di 0 volt esatti perché la tensione non è mai inferiore a valori compresi tra 0,1 e 0,2 volt per un transistore al silicio. Perciò, quando viene applicato un 1 logico all'ingresso del circuito di *figura 3.4a*, non è possibile avere un'uscita di 0 volt. La tensione di uscita si satura a circa 0,2 volt e così attraverso la resistenza del collettore passano poco meno di 5 milliampere. Comunque, la gamma di valori delle tensioni di uscita per uno 0 logico (per la famiglia TTL) va da 0,0 a 0,4 volt; perciò all'uscita si ha ancora uno 0 logico e si dice che il transistore è nel suo stato saturato. Quando la tensione di ingresso non è più elevata della caduta di tensione del transistore tra la base e l'emettitore la corrente di base non fluisce. Perciò la corrente di collettore non scorre, si ha un 1 logico in uscita e si dice che il transistore è spento o è nel suo stato di alta resistenza (milioni di ohm).

Effetto di un corto

Un corto circuito fra l'emettitore e la base di un invertitore a transistori provoca una corrente nulla nel collettore, pone il transistore nel suo stato inattivo e quindi fornisce un 1 logico.

Ora, avendo presente il normale funzionamento di un circuito a transistore, possiamo determinare l'effetto dei corti delle aperture sull'uscita di tale circuito. La *figura 3.4b* mostra un transistore con la base cortocircuitata con l'emettitore. In questo caso tutta la corrente di ingresso scorre attraverso il corto, mentre non passa corrente attraverso la base del transistore. Perciò la corrente del collettore è 0 e il transistore è spento. Quindi la tensione di uscita è alta e l'uscita è fissata al valore logico 1 sia che l'ingresso sia un 1 o uno 0 logico.

Effetto di un'apertura

Una situazione simile si verifica se c'è un'apertura nel circuito di base.

Un'altra situazione che potrebbe mandare l'uscita del transistore ad 1 indipendentemente dal segnale d'ingresso è mostrata in *figura 3.4c*. In questo caso c'è un'apertura nel circuito di base, cosicché, qualunque sia la tensione di ingresso applicata, la corrente di base non può scorrere e perciò nemmeno la corrente di collettore. Anche questa volta l'uscita è bloccata a 1.

Effetto di una resistenza di dispersione

Un altro tipo di guasto in un circuito è mostrato in *figura 3.4d*. In questo caso c'è un'indesiderata ed inaspettata resistenza di dispersione tra la base e la massa. L'applicazione di uno 0 logico all'ingresso spegnerà ancora il transistore e di conseguenza si avrà un 1 logico in uscita. Invece quando viene applicato un 1 logico non tut-

Una dispersione di corrente non desiderata può provocare effetti insoliti.

ta la corrente di ingresso andrà nella base del transistore perché parte di essa passerà attraverso la resistenza di dispersione. Nell'esempio di *figura 3.4d*, la caduta di tensione di 0,6 volt tra la base e l'emettitore provoca il passaggio di una corrente di 15 microampere attraverso la resistenza di dispersione di 40.000 ohm. Poiché la tensione di ingresso in questo caso è di 5,6 volt e si è avuta una caduta di 0,6 volt tra la base e la massa, attraverso la resistenza di 100 kiloohm deve aversi una caduta di 5 volt. Dato che I = V/R = 5/100.000, la corrente è uguale cioè a I = 50 microampere. Sottraendo i 15 microampere della corrente di dispersione dalla corrente di ingresso totale di 50 microampere risulta che soltanto 35 microampere passano nella base del transistore. Questo significa che la corrente di collettore verrà ridotta a 3,5 milliampere in modo tale che la tensione di uscita sia 5 - 3,5, ossia 1,5 volt. La tensione di 1,5 volt non è né uno 0 logico, né un 1 logico, perciò il transistore successivo (che non è mostrato in figura) può essere acceso o spento a seconda delle sue particolari caratteristiche. Quindi la sua uscita può essere sia uno 0 logico che un 1 logico. Inoltre l'uscita può essere differente da istante a istante a seconda della temperatura e dell'alimentazione.

Per esempio, se un normale circuito invertitore come quello osservato in *figura 3.4a* viene accoppiato all'uscita del circuito di *figura 3.4d*, la caduta di tensione attraverso la resistenza di 100 kiloohm potrebbe essere solo 1,5-0,6 volt, cioè 0,9 volt. Perciò la corrente di base sarebbe di 9,0 microampere. Questo provocherebbe una tensione di uscita di 4,5 volt, che sta nella gamma di valori per un 1 logico. Così, il circuito di *figura 3.4a* risponderebbe ad un ingresso di 1,5 volt come se ci fosse uno 0 logico in ingresso. Comunque, un altro invertitore progettato in maniera leggermente diversa potrebbe rispondere ad un ingresso di 1,5 volt come se ci fosse un 1 logico in ingresso. In entrambi i casi, il livello di tensione errato si può trovare procedendo a ritroso nel circuito e misurando la tensione di uscita di ogni transistore nel circuito per determinare dove si manifesta il primo livello errato.

Guasto di un componente

Tutte le situazioni delle *figure 3.4b*, *3.4c* e *3.4d* possono rappresentare guasti esterni o interni al transistore. Se un guasto è interno, esso verrà classificato come guasto di un componente. Se il transistore è un componente discreto, allora il transistore potrà essere sostituito. Se invece esso è un componente di un circuito integrato dovrà essere sostituito l'intero circuito. Tutto quello che un riparatore può fare è determinare attraverso misurazioni e analisi quale componente è difettoso e sostituirlo. Può essere determinato se il guasto sia in un componente o nel cablaggio di solito attraverso un procedimento di eliminazione nel corso delle misurazioni effettuate o sul componente in loco o sul componente fuori dal sistema con una strumentazione adatta alla verifica.

ALTRI TIPI DI TRANSISTORE

Prima di lasciare l'argomento dei guasti nei circuiti a transistore e dei loro effetti sui livelli di tensione dell'uscita del circuito è necessario dire che il transistore bipola-

Figura 3.5. Invertitore a transistore PNP

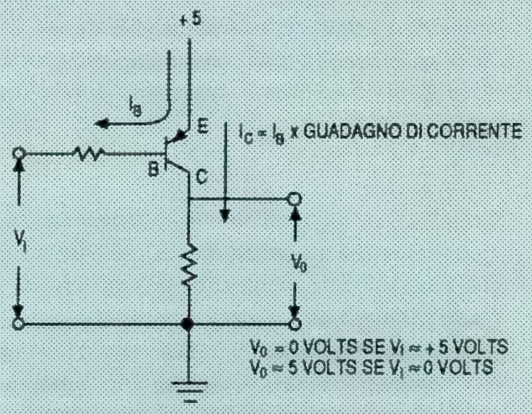
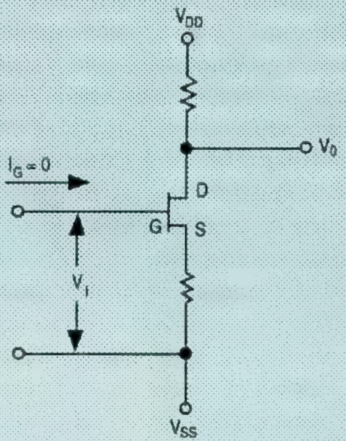

+5
I_B
E
$I_C = I_B$ x GUADAGNO DI CORRENTE
B
C
V_I
V_0
V_0 = 0 VOLTS SE $V_I \approx$ + 5 VOLTS
$V_0 \approx$ 5 VOLTS SE $V_I \approx$ 0 VOLTS

Figura 3.6. Funzionamento di un FET


V_{DD}
V_0
$I_G = 0$
D
G
S
V_I
V_{SS}

$V_{SS} = 0$ V_{DD} = +5 O PIU'

Segnale Logico D'ingresso	V_I	V_0	Segnale Logico D'uscita
0	0	V_{DD}	1
1	5	0	0

a. FET a Canale N

$V_{SS} = 0$ V_{DD} = - 12 VOLTS

Segnale Logico D'ingresso	V_I	V_0	Segnale Logico D'uscita
0	0	- 12	0
1	- 12	0	1

b. FET a Canale P

re NPN mostrato in *figura 3.4* è solo uno dei tipi di transistori che si possono trovare nei circuiti digitali. Altri tipi sono il transistore bipolare PNP, il FET a canale P e il FET a canale N. (Le lettere P ed N si riferiscono al tipo di drogaggio utilizzato nelle regioni del transistore. Le lettere FET sono un'abbreviazione per Field Effect Transistor, Transistore ad effetto di campo). I circuiti fondamentali e le caratteristiche funzionali di questi dispositivi sono mostrati in *figura 3.5* e in *figura 3.6*.

PNP

Il transistore bipolare PNP mostrato nello schema in *figura 3.5* ha caratteristiche funzionali simili a quelle del transistore NPN mostrato in *figura 3.4a*. Confrontando le due figure si vede che la principale differenza sta nelle polarità delle tensioni collegate all'emettitore e al collettore e nelle direzioni del flusso di corrente. Si noti che l'emettitore del PNP è collegato al polo positivo dell'alimentatore, mentre il collettore è collegato a massa, cioè al polo negativo dell'alimentatore, attraverso la resistenza di collettore.Il circuito di base è collegato alla tensione di ingresso e la tensione di uscita è prelevata dal collettore del transistore rispetto alla massa.

Il transistore PNP funziona come il transistore NPN, ad eccezione del fatto che le polarità della tensione sono invertite e il passaggio di corrente avviene nel verso opposto rispetto a quello che si ha in un transistore NPN.

Proprio come nel transistore NPN, se il terminale di base del transistore PNP è collegato allo stesso potenziale del terminale di emettitore non può fluire la corrente di base e quindi nemmeno la corrente di collettore. Di conseguenza la tensione all'uscita sarà di 0 volt perché non c'è caduta di tensione attraverso le resistenza di collettore, e così l'estremità superiore del resistore sarà allo stesso livello di tensione dell'estremità inferiore cioè a 0 volt. Perciò una tensione di ingresso di 5 volt (corrispondente ad un 1 logico) darà come risultato una tensione di uscita di 0 volt (corrispondente ad uno 0 logico) dando luogo così ad un'inversione logica.

Se la tensione di ingresso è uno 0 logico, ad esempio è di 0 volt, si ha passaggio di corrente attraverso il diodo emettitore-base polarizzato direttamente e attraverso la resistenza di base verso il collegamento di 0 volt all'ingresso. La corrente che fluirà nel collettore sarà uguale a questa corrente di base moltiplicata per il guadagno di corrente del transistore. Se il transistore è saturato, cioè la tensione tra l'emettitore e il collettore è di alcuni decimi di volt, la tensione di uscita sarà la caduta di tensione sulla resistenza di collettore, cioè circa 5 volt. Così, l'applicazione di uno 0 logico (0 volt) all'ingresso produrrà un livello logico 1 in uscita (circa 5 volt). I guasti nei circuiti PNP si potranno individuare utilizzando le stesse tecniche trattate precedentemente per i circuiti NPN.

FET

Un FET viene controllato dalla tensione invece che dalla corrente, come nel caso di un transistore

In *figura 3.6* è illustrato il funzionamento del FET. La differenza più importante tra il transistore bipolare e il FET è che quest'ultimo è sensibile alla tensione invece che alla corrente. La sua corrente di ingresso è 0, indipendentemente dalla tensione di ingresso. (L'ingresso del FET si chiama gate invece di base. Il source e il drain del FET corrispondono rispettivamente all'emettitore e al collettore del transistore bipolare). Se viene collegato un ohmetro tra la base e l'emettitore di un transistore bipolare con bipolare. I circuiti a FET hanno una resistenza d'ingresso molto elevata.

polarità tale da polarizzare direttamente la giunzione il valore della resistenza misurato sarebbe piccolo. Invertendo la polarità dei collegamenti, l'ohmetro indicherà un valore della resistenza più elevato. Se questa stessa misurazione viene compiuta con qualunque polarità dalla porta di un FET verso il terminale del pozzo o della sorgente, sarebbe normale avere sull'ohmetro l'indicazione di una resistenza molto elevata o infinita. Questo avviene per il modo in cui è costruito il FET. Anche così, il FET risponderà ad una tensione di ingresso di un dato livello per produrre una particolare tensione di uscita.

FET a canale N

I FET a canale N hanno una tensione positiva applicata al drain e una tensione nulla o negativa applicata al gate quando funzionano come un invertitore.

Nel dispositivo a canale N, V_{DD} è una tensione positiva, mentre V_{SS} è 0 volt (massa). L'applicazione di una tensione dello stesso livello di V_{SS} al gate non produrrebbe tensione tra gate e source; perciò non scorrerebbe corrente nel drain e la tensione di uscita sarebbe al livello di V_{DD}, poiché non si avrebbe alcuna caduta di tensione sulla resistenza. In questo caso un ingresso di 0 volt (livello logico 0) produrrebbe una tensione positiva V_{DD} in uscita (livello logico 1) e sarebbe realizzata così un'inversione logica. Analogamente, se viene applicata al gate una tensione prossima al valore V_{DD} (1 logico), si ottiene una tensione tra gate e source sufficiente a provocare il flusso massimo di corrente nel drain, il che riduce la tensione in uscita a valori vicini al livello V_{SS} o a 0 volt. Perciò, l' ingresso di un 1 logico produce l'uscita di uno 0 logico. Gli effetti di questi ingressi sulle uscite sono mostrati in *figura 3.6*.

FET a canale P

I FET a canale P hanno una tensione negativa applicata al drain e una tensione di 0 volt o negativa applicata al gate quando funzionano da invertitore.

Il FET a canale P funziona con tensioni di polarità opposta rispetto a quelle del FET a canale N; perciò esso richiede una tensione V_{SS} che sia positiva rispetto a V_{DD}. Per esempio, se V_{SS} è 0 volt allora V_{DD} dovrà essere un valore negativo di tensione. Se la tensione di ingresso è al livello V_{SS}, cioè 0 volt, non c'è tensione tra gate e source e quindi non passa corrente di drain e la tensione di uscita è al livello V_{DD}. Prendendo la tensione più positiva (in questo caso 0 volt) come livello di 1 logico e la tensione più negativa (in questo caso il livello V_{DD}) come livello di 0 logico, si può vedere che l'ingresso di un 1 logico produrrà uno 0 logico in uscita. Analogamente, se V_i è al livello V_{DD} (una tensione negativa) si ha una significativa tensione tra gate e source della polarità corretta per fornire la massima corrente di drain, cosicché la tensione di uscita è vicina al livello V_{SS} di 0 volt. Perciò l'ingresso di uno 0 logico (-12 volt in questo caso) produrrà l'uscita di un 1 logico (0 volt).

Si noti che in tutti gli esempi precedenti si è usata la stessa convenzione per cui l'1 logico rappresenta la tensione più positiva e lo 0 logico quella più negativa. Questa logica viene detta logica "positiva". La particolarità di questo circuito è che la massa o 0 volt è un 1 logico piuttosto che una tensione positiva (+), e una tensione negativa (-) è uno 0 logico piuttosto che la massa o 0 volt. In altre parole, la logica è la stessa, ma il diverso punto di riferimento di massa fa sì che siano tensioni differenti a produrre quella logica.

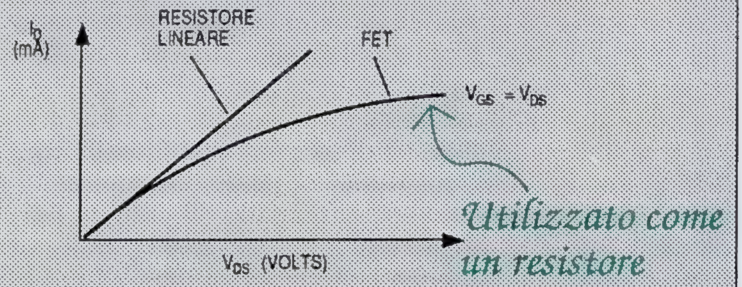
Se un circuito è formato da due o più tipi di transistori mostrati nelle *figure 3.4*, *3.5* e *3.6*, potrebbero essere necessari circuiti di interfaccia posti tra di essi per rendere compatibili i livelli di tensione di 0 logico ed 1 logico che si verificano nei differenti invertitori. Un tale circuito di interfaccia sarebbe indubbiamente necessario per interfacciare il circuito del transistore NPN di *figura 3.4* con il circuito FET a canale P di *figura 3.6*.

Uso di un transistore FET come resistore

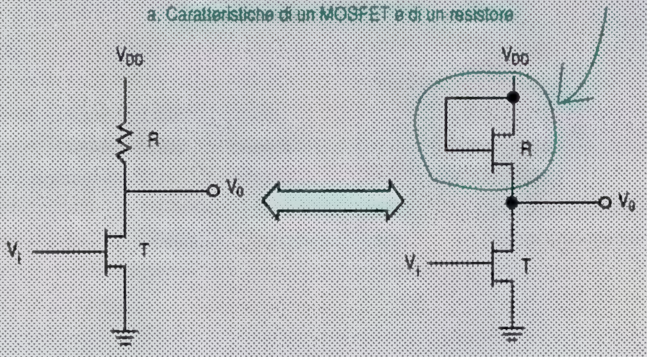
Gli invertitori discussi nelle *figure 3.4*, *3.5* e *3.6* utilizzano una resistenza posta tra il collettore, o drain, del transistore e l'alimentatore in corrente continua (V_{cc} o V_{dd}). Nei circuiti integrati è spesso troppo costoso fare una resistenza con valore in ohm molto grande per la quantità di spazio che il resistore occupa sulla superficie del chip di silicio. È molto più economico fare dei transistori sul silicio. Fortunatamente, un transistore può comportarsi come un resistore in certe condizioni funzionali.
La relazione tra I_d e V_{ds} per un transistore MOSFET è mostrata in *figura 3.7a* per il caso in cui il terminale di gate è collegato col terminale di drain. Questo provoca una tensione V_{gs} relativamente grande cosicché il transistore è attivo e conduce. In *figura 3.7a* viene mostrato anche il grafico della corrente in funzione della tensione per

Nei progetti dei circuiti integrati MOS un transistore MOS a effetto di campo viene utilizzato come resistenza elevata perché è più economico integrarlo.

Figura 3.7. Utilizzo di un MOSFET come resistore



a. Caratteristiche di un MOSFET e di un resistore



b. Circuiti Invertitori Equivalenti

un resistore. Confrontando le curve, si può vedere che per valori piccoli di V_{DS} il FET si comporta come un resistore. Anche per valori elevati di V_{DS} il FET si comporta ancora come un resistore, ma la resistenza varia al variare della tensione applicata. Perciò, la resistenza del FET non è lineare, ma per le applicazioni considerate nei circuiti di *figura 3.6* esso lavora come se si avesse un resistore lineare. In *figura 3.7b* è mostrato il resistore lineare sostituito dal FET che è fisicamente più piccolo (per quanto riguarda la superficie di silicio), ma che ha un valore di resistenza equivalente. Questo è il progetto standard utilizzato nei circuiti integrati MOS. Il funzionamento del circuito è uguale a quello dei circuiti a transistori MOS mostrati in *figura 3.6*. I guasti nei circuiti che usano FET a canale P o FET a canale N possono essere individuati utilizzando gli stessi metodi discussi per il transistore bipolare NPN di *figura 3.4*, a parte il fatto che non può essere misurata la corrente di gate. Gli strumenti di misura utilizzati per la ricerca dei guasti nei circuiti a transistore - il voltmetro, l'amperometro, l'ohmetro e la sonda logica - saranno considerati più dettagliatamente nel seguente paragrafo.

TECNICHE DI MISURAZIONE

Gli strumenti di test trattati in questo capitolo forniscono informazioni riguardo ai livelli di tensione, di corrente o di resistenza in un circuito. Essi indicano il livello della quantità misurata tramite il movimento di un ago su una scala metrica graduata (misurazione analogica), o tramite un display digitale che segnala l'effettivo valore numerico (misurazione digitale), o accendendo o spegnendo un dispositivo che emette luce (sonda logica). Sono tutti utili per determinare la posizione e la natura dei guasti in un sistema digitale.

Multimetri

Uno strumento di test di uso generale in grado di misurare tensione, corrente e resistenza è il multimetro. Può essere sia di tipo analogico, come quello mostrato in *figura 3.8a*, che di tipo digitale, come mostrato in *figura 3.8b*. Il tipo analogico è generalmente meno costoso del tipo digitale, ma non è così comodo da utilizzare e spesso non è così accurato come il tipo digitale.

Scelta della funzione e del fondo scala

I multimetri misurano tensione, corrente e resistenza e possono visualizzare queste letture sia in forma analogica che in forma digitale.

Entrambi i tipi di multimetro dispongono di commutatori di comando per scegliere la funzione (tensione, corrente o resistenza) e la gamma di variazione o scala. Perciò, se devono essere misurate tensioni in corrente continua di valori che ci si aspetta essere al di sotto di 10 volt, i commutatori dovranno essere regolati in modo tale da scegliere la funzione di tensione continua e una scala che va da 0 a 10 volt. Per uno strumento analogico, la scala che corrisponde alla gamma scelta si legge sul quadrante dello strumento in corrispondenza della posizione in cui si trova l'ago. Per uno strumento digitale, apparirà sul display un numero compreso tra 0,000 e 9,999 per indicare direttamente il reale valore della tensione misurata. Questo è uno dei van-

Scale e gamme possono essere cambiate facilmente tramite gli interruttori del selettore.

taggi dello strumento digitale. Un altro vantaggio è costituito dal "autoranging", che è un modo di funzionamento nel quale lo strumento stesso sceglie la scala più conveniente per la lettura migliore della tensione di ingresso. Perciò lo strumento digitale compie tutto il lavoro – l'operatore deve semplicemente collegare l'ingresso.

Polarità

Assicurarsi che le polarità delle sonde di misurazione di un qualsiasi strumento utilizzato per il test concordino perfettamente con le polarità dei punti che si stanno verificando nel circuito.

Entrambi gli strumenti hanno delle boccole (jack) sul contenitore nelle quali possono essere inserite le sonde di misurazione. Le boccole portano il segno + e - e/o sono di colore rosso (per il +) e nero (per il -). Per misurare la tensione e la corrente si collega la sonda rossa col punto più positivo del circuito e la sonda nera con quello più negativo. Ad eccezione del circuito a canale P di *figura 3.6*, tutti i circuiti che sono stati mostrati hanno come punto più negativo la massa, ma in molti altri circuiti, così come nel circuito a canale P, la massa è il punto più positivo.

ATTENZIONE: spegnere sempre l'alimentazione del circuito prima di effettuare misure di resistenza.

Per misure di resistenza, le due sonde verranno messe ai capi del circuito di cui si deve misurare la resistenza e sarà lo strumento stesso ad applicare al circuito una tensione. Per i resistori la polarità (+ o -) di questa tensione non ha importanza, ma ne ha per i transistori e i diodi. In alcuni strumenti, la boccola rossa è positiva come c'è da aspettarsi, ma in altri è quella nera ad essere positiva.

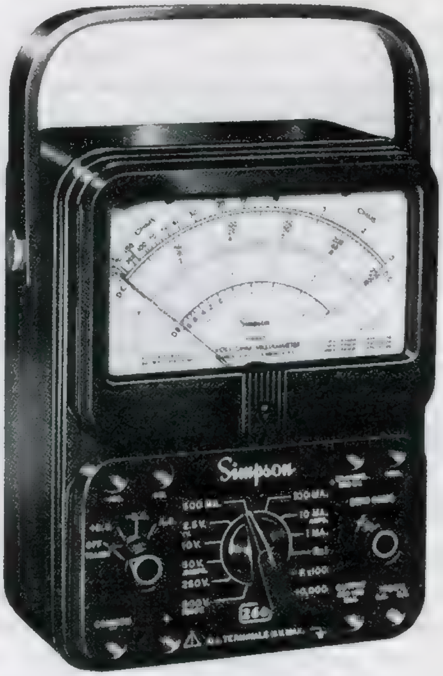
Resistenza di ingresso

Quando si stanno raccogliendo misurazioni in un circuito, assicurarsi che la strumentazione per il test non perturbi il circuito su cui si svolgono le misurazioni, poiché ciò potrebbe influire negativamente sul funzionamento di quest'ultimo.

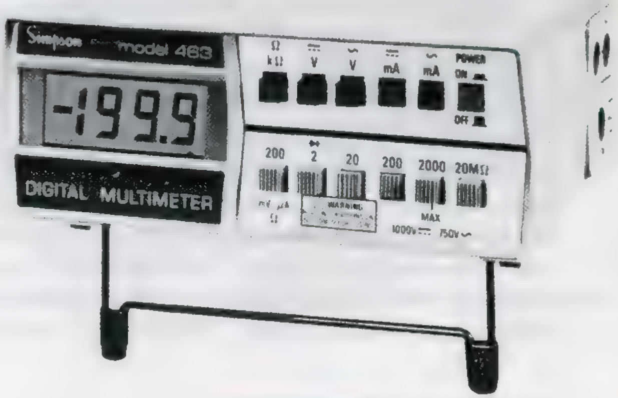
Quando si sta effettuando una qualsiasi misurazione con un qualsiasi strumento, lo strumento scelto deve essere progettato in modo tale che il suo inserimento non alteri il funzionamento del circuito da misurare. Per misure in corrente continua la resistenza di ingresso è il parametro dello strumento che può facilmente influire sul funzionamento del circuito. La *figura 3.9* illustra l'effetto della resistenza di ingresso di uno strumento sul funzionamento di un circuito e i valori misurati della tensione. La *figura 3.9a* mostra un voltmetro che ha una resistenza di ingresso, relativamente elevata, di 1 megaohm cosicché esso assorbirà pochissima corrente dal circuito su cui si sta eseguendo la misurazione. (Alcuni strumenti hanno una resistenza di ingresso maggiore di 10 megaohm).

Quanto lo strumento influisca sul circuito e sulla misurazione della tensione viene determinato dal rapporto fra la resistenza dello strumento e la resistenza del circuito a cui è collegato lo strumento stesso. La combinazione in parallelo di queste due resistenze determina la resistenza equivalente del circuito quando lo strumento è collegato. La formula per calcolare la resistenza equivalente di due resistenze in parallelo è mostrata in *figura 3.9b*. La resistenza equivalente (R_{eq}) di due resistenze (R_1 ed R_2) in parallelo viene calcolata dividendo il prodotto delle due resistenze per la somma delle due resistenze. Nel circuito di *figura 3.9a* la resistenza del circuito di 10 kiloohm (R_2) in parallelo con la resistenza dello strumento di 1 megaohm determina una resistenza equivalente di 9901 ohm. Perciò la tensione ai capi di R_2 viene indicata es-

Figura 3.8. Multimetri *(Per gentile concessione della Simpson Electric Company, Elgin, Illinois)*


Simpson
260

a. Strumento Analogico


Simpson model 463
-199.9
DIGITAL MULTIMETER
kΩ
V
V
mA
mA
POWER
ON
OFF
200
2
20
200
2000
20MΩ
MAX
1000V
750V

b. Strumento Digitale

sere circa 2,49 volt invece del valore effettivo di 2,50 volt che si ha quando lo strumento non è collegato. Quindi l'atto di misurazione della tensione ha alterato la tensione di circa 0,01 volt, ma questo è un errore inferiore allo 0,5% della misurazione, la qual cosa è accettabile per la maggior parte delle esigenze.

Se un voltmetro con una resistenza interna di soli 10.000 ohm viene utilizzato nella misurazione, come mostrato in *figura 3.9c*, la resistenza equivalente alle due resistenze in parallelo di 10.000 ohm è di 5000 ohm. Ora la tensione indicata dallo strumento è di 1,67 volt, il che comporta un errore maggiore di oltre il 30%. A parte l'errore di misurazione, il funzionamento di un circuito potrebbe essere alterato fino al punto da sembrare affetto da guasti quando invece non lo è.

Ci si riferisce a questa condizione chiamandola "perturbazione della misura dovuta allo strumento" ed è necessario evitarla durante la ricerca di guasti. In generale, per evitare di "perturbare" il circuito e di alterarne le tensioni, la resistenza di ingresso o impedenza dello strumento di misura deve essere almeno 100 volte la resistenza che lo strumento deve misurare. Ciò vale per tutte le apparecchiature di misurazione, comprese le semplici sonde logiche che verranno trattate nel seguente paragrafo.

Sonde logiche

La sonda logica è un tipo di voltmetro che fornisce informazioni circa il livello logico di una particolare linea di segnale. Se il livello è ad 1, la luce si accende; se è a 0, la luce resta spenta.

Una sonda logica è una specie di voltmetro. Esistono diversi tipi di sonde logiche ed alcuni dispositivi correlati come i generatori di impulsi logici che sono disponibili per aiutare ad analizzare il circuito. Alcuni di questi dispositivi sono mostrati in *figura 3.10*. Una sonda logica fornirà al riparatore le informazioni relative ai segnali su un linea. La clip logica (sonda a tenaglia) fornirà invece le informazioni relative ai segnali su 16 linee. Quando un segnale è al livello logico 1, la luce sulla sonda logica si accenderà. Quando invece il segnale è al livello logico 0 la luce si spegnerà. Perciò il riparatore avrà una buona indicazione circa lo stato di un determinato percorso del segnale.

Clip logica

In pratica, una clip logica con 16 contatti sono 16 sonde logiche in un solo contenitore. Essa permette il collegamento a tutti i piedini di un circuito integrato a 16 piedini contemporanea–mente.

Un secondo tipo di sonda logica è mostrato in *figura 3.10*. Talvolta essa viene chiamata clip logica poiché è una sonda su cui è montata una clip con 16 contatti che si collegano ai 16 piedini del circuito integrato. Ci sono anche 16 spie luminose, una per ciascun piedino, cosicché i segnali su tutti e 16 i piedini possono essere controllati contemporaneamente. Una sonda di questo tipo è in genere sensibile all'ampiezza del segnale mentre non è in grado di rilevare i livelli istantanei di 1 che si verificano in impulsi digitali rapidi. Collegando tale clip al circuito e provvedendo che siano note le condizioni di ingresso, l'intero funzionamento del circuito può essere osservato abbastanza facilmente attraverso il comportamento delle spie luminose.

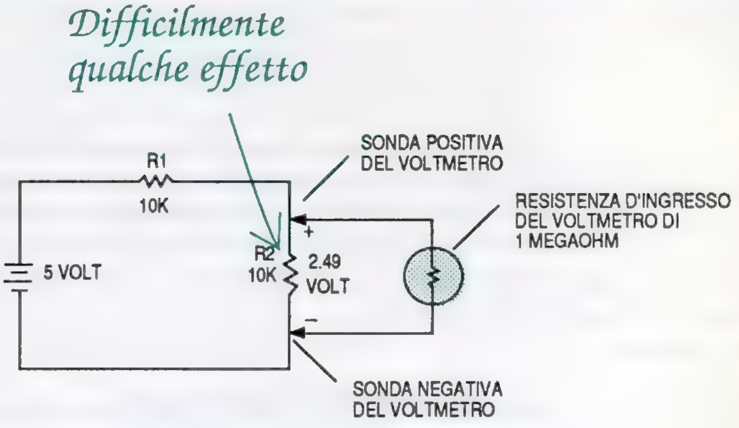
Impulsatore logico

Per inserire impulsi di segnale in un circuito al fine di effettuare una

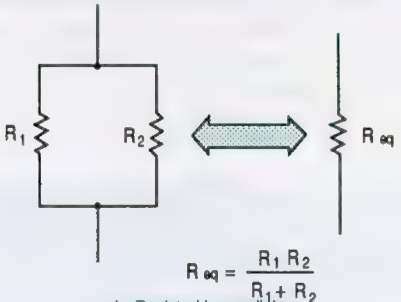
Un dispositivo che può essere utilizzato congiuntamente alla sonda logica o alla clip logica è l'impulsatore logico mostrato in *figura 3.10*. Esso produce impulsi che vengono inseriti o iniettati nel circuito nel punto di contatto.

Generalmente, il riparatore ha sotto controllo quando e quanto spesso gli impulsi

Figura 3.9. Effetto della resistenza d'ingresso del misuratore sulle tensioni misurate

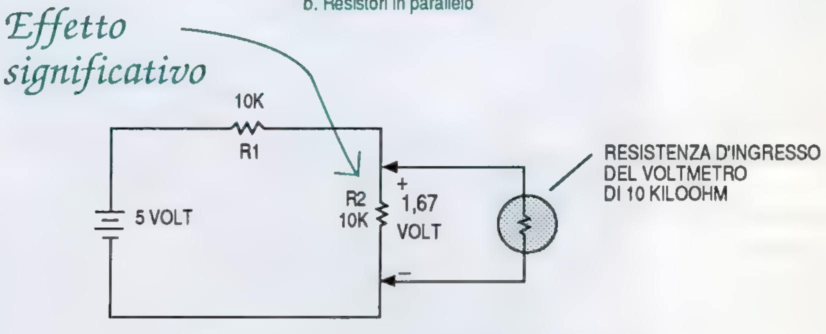

Difficilmente qualche effetto
R1
10K
5 VOLT
R2
10K
+
2.49
VOLT
−
SONDA POSITIVA DEL VOLTMETRO
RESISTENZA D'INGRESSO DEL VOLTMETRO DI 1 MEGAOHM
SONDA NEGATIVA DEL VOLTMETRO

a. Effetto di un voltmetro con Resistenza di 1 Megaohm



$$R_{eq} = \frac{R_1 R_2}{R_1 + R_2}$$

b. Resistori in parallelo


Effetto significativo
10K
R1
5 VOLT
R2
10K
+
1,67
VOLT
−
RESISTENZA D'INGRESSO DEL VOLTMETRO DI 10 KILOOHM

c. Effetto di un voltmetro con Resistenza d'Ingresso di 10 Kiloohm rilevazione viene utilizzato un impulsatore logico.

vengono iniettati tramite un interruttore a pulsante. Solitamente viene generato un impulso ogniqualvolta il pulsante viene premuto e rilasciato. Premendo e rilasciando successivamente il pulsante può essere iniettato un treno di impulsi di frequenza relativamente bassa nel circuito. In alcuni impulsatori un treno di impulsi viene prodotto semplicemente tenendo premuto il pulsante.
Utilizzando l'impulsatore congiuntamente alla sonda logica o alla clip logica, il passaggio degli impulsi generati dall'impulsatore attraverso il circuito viene indicato dall'intermittenza delle spie luminose sulla sonda logica o sulla clip logica. Un test combinato di questo genere permetterà al riparatore di seguire gli impulsi lungo tutti i percorsi di un circuito in modo da determinare quali di essi funzionano adeguatamente e quali no.

METODI DI RICERCA DEI GUASTI

Il procedimento per determinare la posizione di un guasto in un sistema implica che si debba ragionare sul funzionamento del sistema per isolare dapprima la generica unità funzionale che non lavora correttamente. Questo procedimento di ragionamento può essere aiutato attraverso prove raccolte da una serie di misurazioni ben pianificate. Una volta isolato il guasto in una precisa unità funzionale, esso può essere individuato con precisione seguendo lo stesso procedimento di ragionamento e lo stesso piano di misurazioni dettagliate all'interno dell'unità funzionale. In tutti que-

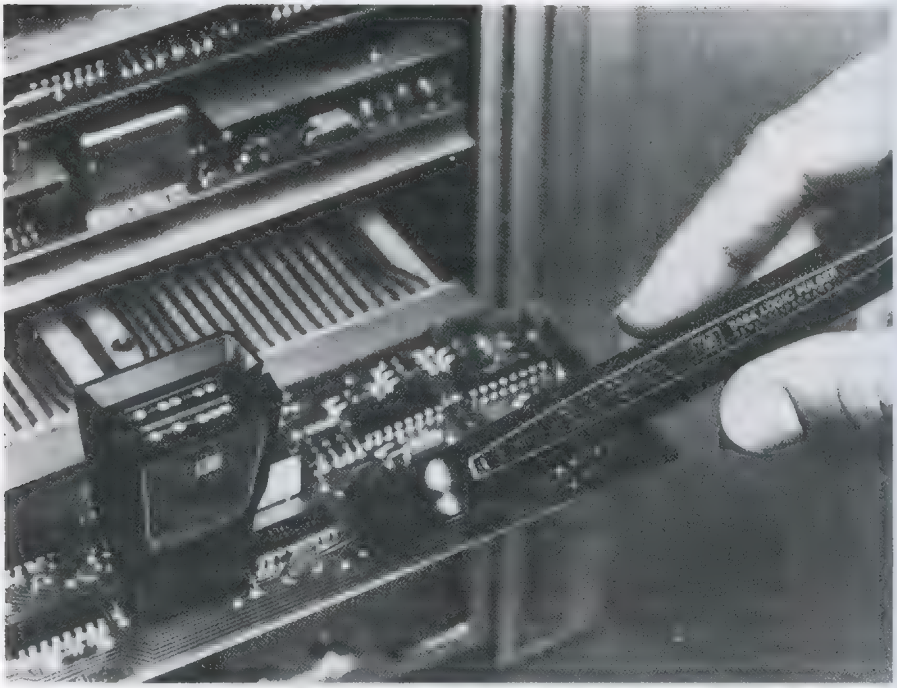


Figura 3.10. Impulsatore logico e clip logica *(Per gentile concessione della Hewlett Packard Co.)*

sti piani di misurazione e analisi di sistemi, il riparatore deve pensare al sistema come ad un certo numero di percorsi in serie del segnale, ognuno dei quali deve funzionare correttamente perché anche il sistema funzioni correttamente.

Analisi funzionale

Una perfetta conoscenza di come un sistema lavora quando sta funzionando correttamente è un mezzo molto valido per osservare il funzionamento di un sistema guasto.

Un esempio del diagramma di flusso funzionale valido in un qualunque sistema è mostrato per un caso generale in *figura 3.11a*. Ciascun sistema deve avere una funzione di ingresso, una funzione di elaborazione e una funzione di uscita. Se una di queste funzioni si guasta, l'intero sistema non funzionerà correttamente. Osservare il funzionamento del sistema può fornire un gran numero di informazioni circa quale funzione di ingresso o funzione di elaborazione o funzione di uscita sia la causa del problema.

Un esempio più specifico della natura funzionale di un sistema elettronico, un televisore a colori, è mostrato in *figura 3.11b*. Il sistema descritto non è un sistema digitale, ma le tecniche utilizzate nella ricerca del guasto sono le stesse sia per sistemi analogici che per sistemi digitali. Talvolta i guasti possono essere causati da un difetto in una funzione non delle più evidenti; comunque, nella trattazione seguente verranno considerate solo quelle più evidenti.

Funzionamento di un televisore

Viene utilizzato come esempio il funzionamento di un ricevitore TV.

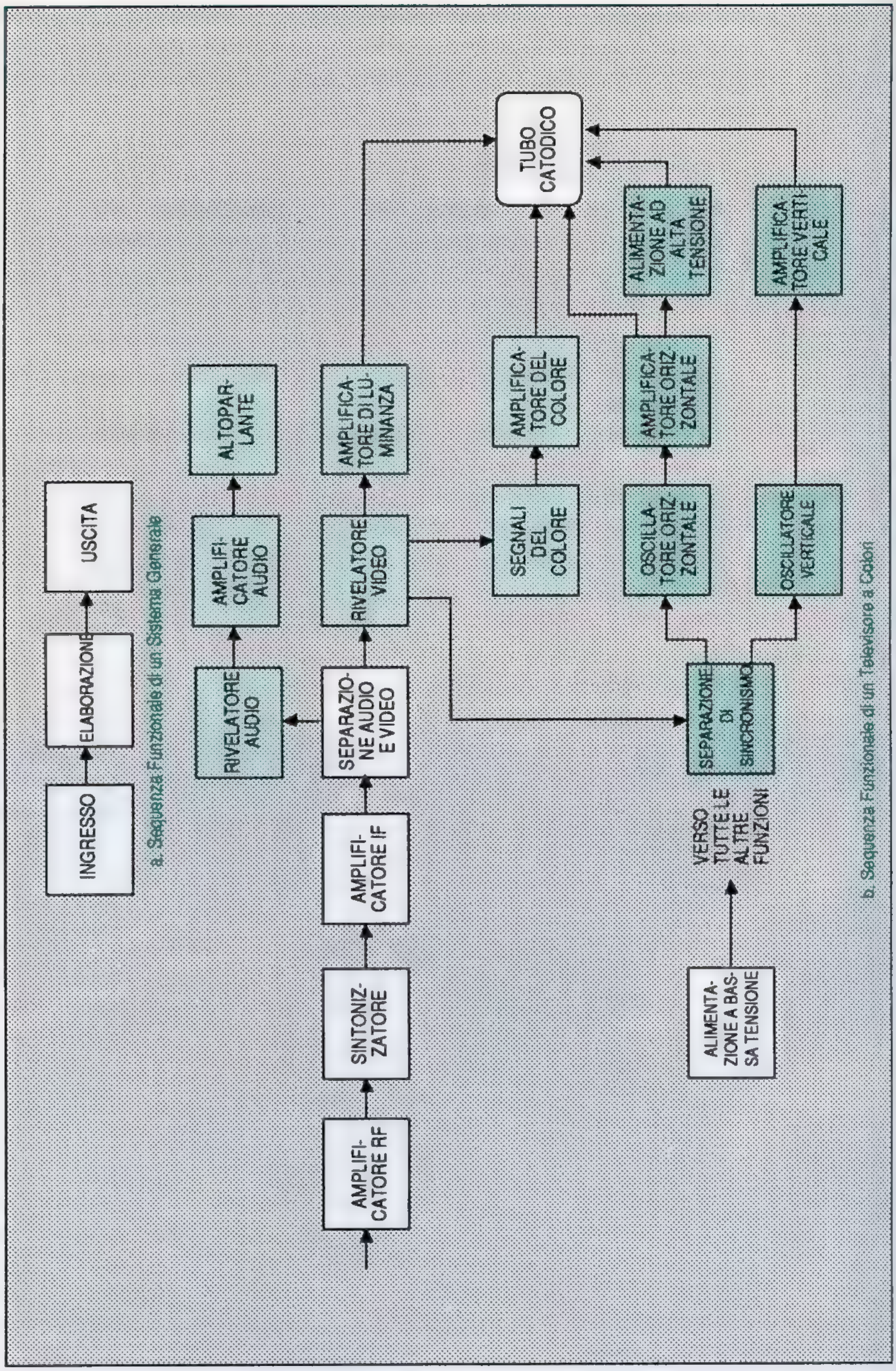
I circuiti o le funzioni di ingresso sono quelli di amplificazione e sintonia. L'amplificatore RF porta il debole segnale di antenna ad un livello che può essere elaborato dalla sezione di sintonia per scegliere un canale televisivo tra tutti i canali a disposizione. Questa funzione viene seguita da speciali amplificatori accordati, chiamati amplificatori IF, i quali scartano i segnali non desiderati ed aumentano l'intensità del segnale scelto.

L'operazione seguente è quella di separare il segnale audio da quello video. Da questo punto in poi nel sistema il percorso audio sarà indipendente dalle altre funzioni. Il segnale audio procede verso un rivelatore audio che recupera i segnali di frequenza audio dalla sua portante. L'audio viene amplificato al livello richiesto per far funzionare gli altoparlanti.

Il rivelatore video recupera il segnale video dalla sua portante. I segnali d'immagine vengono separati in segnale di luminanza (che determina le parti bianche e quelle nere di un'immagine) e in segnali di burst e cromatismo (segnali che contengono informazioni sul colore). I segnali di definizione del colore vengono elaborati e amplificati per produrre i segnali di controllo del colore e vengono inviati insieme al segnale di luminanza ai cannoni elettronici del tubo catodico per riprodurre a colori l'immagine originale.

Il segnale di sincronizzazione viene inviato per controllare gli oscillatori orizzontali e verticali. Questi circuiti generano forme d'onda le quali fanno sì che i fasci di elettroni generati dai cannoni elettronici nel tubo catodico siano deflessi orizzontalmente e verticalmente sullo schermo per formare un'immagine perfettamente rettangolare. Questi segnali sono modulati, amplificati e trasmessi alle bobine di deflessione del tubo catodico.

Figura 3.11. Relazioni funzionali di un sistema


INGRESSO
ELABORAZIONE
USCITA
AMPLIFICATORE RF
SINTONIZZATORE
AMPLIFICATORE IF
SEPARAZIONE AUDIO E VIDEO
RIVELATORE AUDIO
AMPLIFICATORE AUDIO
ALTOPARLANTE
RIVELATORE VIDEO
AMPLIFICATORE DI LUMINANZA
SEGNALI DEL COLORE
AMPLIFICATORE DEL COLORE
TUBO CATODICO
SEPARAZIONE DI SINCRONISMO
OSCILLATORE ORIZZONTALE
AMPLIFICATORE ORIZZONTALE
ALIMENTAZIONE AD ALTA TENSIONE
OSCILLATORE VERTICALE
AMPLIFICATORE VERTICALE
ALIMENTAZIONE A BASSA TENSIONE
VERSO TUTTE LE ALTRE FUNZIONI

a. Sequenza Funzionale di un Sistema Generale

b. Sequenza Funzionale di un Televisore a Colori

Ricerca dei guasti in un televisore

Quando si verifica un guasto, ragionando sulle funzioni dei sottosistemi e sul malfunzionamento causato dal guasto, dovrebbe risultare evidente il sottosistema che è probabilmente guasto. Tecniche di ricerca dei guasti più dettagliate vengono richieste per individuare l'esatta causa del problema.

Persino con questa breve descrizione è possibile analizzare i sintomi di quasi tutti i guasti e sviluppare una lista delle posizioni in cui con più probabilità si sviluppa il guasto all'interno del sistema. Per esempio, se il problema riscontrato è quello dell'assenza di audio (e non ci sono nemmeno rumori) o dell'immagine, e lo schermo è nero, il riparatore si trova davanti l'improbabile possibilità che due o più percorsi indipendenti del circuito si siano guastati contemporaneamente, o che tutti i circuiti nel sistema abbiano perso la loro alimentazione. L'ultimo caso è quello più probabile e quindi, per prima cosa, andrebbe esaminato l'alimentatore a bassa tensione. Se l'apparecchio non produce suoni, ma fornisce una buona immagine a colori, il guasto deve trovarsi da qualche parte tra la funzione di separazione del suono e gli altoparlanti. Non è possibile localizzare ulteriormente il problema senza effettuare alcune misurazioni sul sistema o senza svolgere altre ricerche del guasto come la sostituzione parziale o la misurazione di un componente.

Nel caso in cui ci siano l'audio e le immagini in bianco e nero (non a colori), il guasto sarà probabilmente nelle sezioni che hanno a che fare con i segnali del colore, come il filtro passa-banda di cromatismo o le sezioni dell'oscillatore di burst. Ancora una volta sono necessarie più informazioni per determinare la causa esatta del problema. Se non c'è né l'audio né l'immagine del programma ma lo schermo è illuminato uniformemente con uno sfondo bianco e si odono rumori dall'altoparlante, c'è l'alimentazione del sistema e il guasto deve trovarsi in funzioni che sono comuni sia all'audio che al video; quelle cioè tra l'antenna (supponendo sia presente un segnale proveniente dall'antenna) e i rivelatori audio e video. Come ultimo esempio si supponga che ci sia l'audio del programma e che l'immagine sia a colori, ma che il quadro stia scorrendo verticalmente. I principali elementi funzionali che comportano il collocamento verticale e orizzontale dell'immagine sono il separatore di sincronismo e gli oscillatori e amplificatori verticali e orizzontali. Esistono altre possibili cause più remote, ma il riparatore probabilmente comincerà col rilevare misurazioni sui circuiti dell'oscillatore e dell'amplificatore verticale per localizzare il guasto.

Indipendentemente dal sistema in esame, si deve compiere questo ragionamento partendo dalle funzioni di ingresso, attraverso quelle di elaborazione fino a quelle di uscita, per avere un'idea di quale di queste funzioni è guasta. Una volta determinata la funzione e localizzato il guasto, diviene allora possibile utilizzare le diverse tecniche di ricerca dei guasti ed effettuare alcune misurazioni che porteranno ad individuare il guasto vero e proprio. Due delle strategie di ricerca dei guasti più efficaci sono l'ispezione visiva e la sostituzione di un componente.

Ispezione visiva

Spesso il guasto di un sottosistema può essere determinato semplicemente ispezionando visivamente il sottosistema.

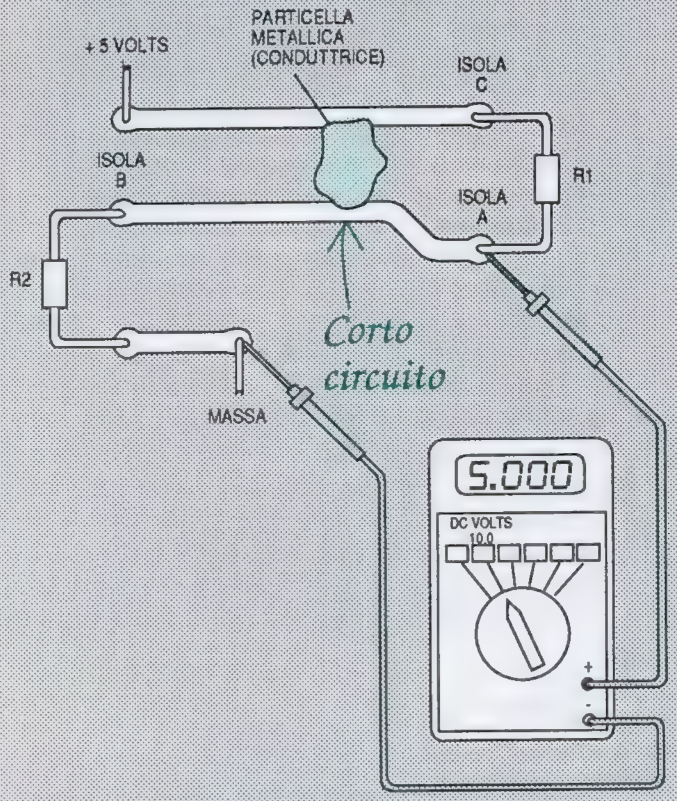
Una volta messo in relazione il guasto di un sistema con un elemento funzionale o una parte del sistema attraverso procedimenti di analisi e misurazioni, è possibile che esso possa essere individuato visivamente; cioè, semplicemente guardando il cablaggio e i componenti del circuito. Nel caso del televisore a colori, per esempio, se l'alimentatore a bassa tensione è considerato responsabile del guasto poiché lo schermo è nero e non c'è audio, e se vengono utilizzate valvole, guardare le valvole per vedere se i filamenti sono incandescenti indicherà se il problema sta nell'alimentazione del filamento o altrove. Analogamente, se il guasto del televisore a colori risiede nei circuiti di generazione del colore (circuiti di burst e di crominanza), dovrebbe essere esaminata quella regione del circuito stampato o il cablaggio del telaio per cercare fili o zoccoli staccati, fili in corto e così via.

Funzionamento intermittente

Se i guasti sono intermittenti, rimuovere uno alla volta i componenti o i collegamenti può aiutare ad individuare il problema.

In un caso del genere (mancanza di colore), l'autore ha esaminato visivamente l'area comprendente questi circuiti senza trovare alcun guasto evidente. Muovendo materialmente fili, valvole, zoccoli e altri componenti con una bacchetta di legno, si scoprì che il colore appariva quando veniva mossa una delle bobine. I collegamenti elettrici della bobina si erano deteriorati col tempo e producevano un collegamento aperto intermittente (spento e acceso) in quella parte del circuito. Saldando i col-

Figura 3.12. Rivelazione visiva ed elettrica dei guasti fisici



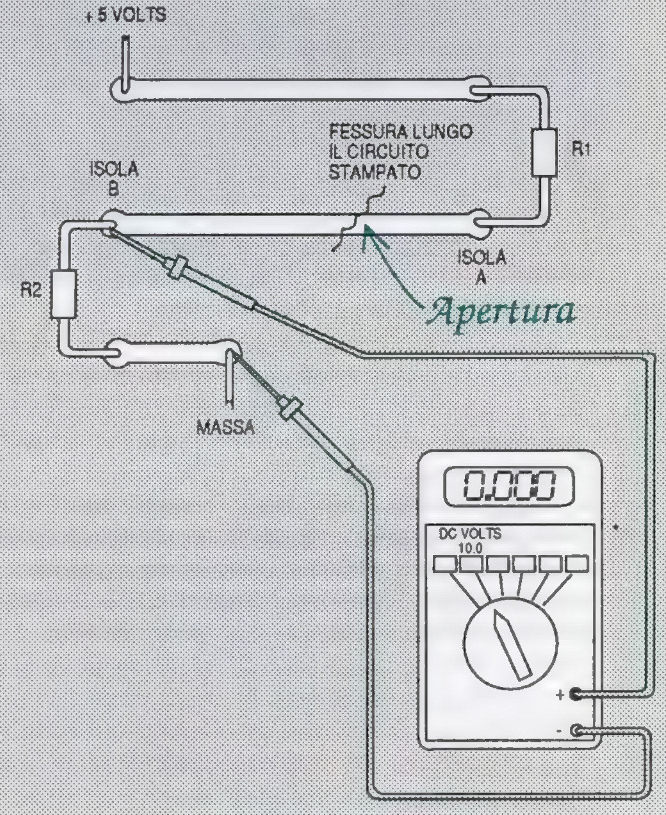
a. Causa Fisica del Guasto di Figura 3.1c legamenti della bobina, il problema venne risolto e il colore fu ripristinato senza nemmeno compiere una misurazione della tensione o controllare alcun componente nel sistema.

Altri esempi dei tipi di problema che possono essere trovati mediante l'ispezione visiva dell'unità sospetta del sistema sono mostrati in *figura 3.12*. Nelle situazioni mostrate in questa figura, si sospetta o un corto circuito o un'apertura nel circuito in seguito all'analisi del comportamento del sistema oppure la presenza di uno di questi guasti è stata confermata dalle misurazioni di tensione e resistenza. Il problema è localizzare fisicamente il corto o l'apertura, se possibile visivamente

Corto circuito

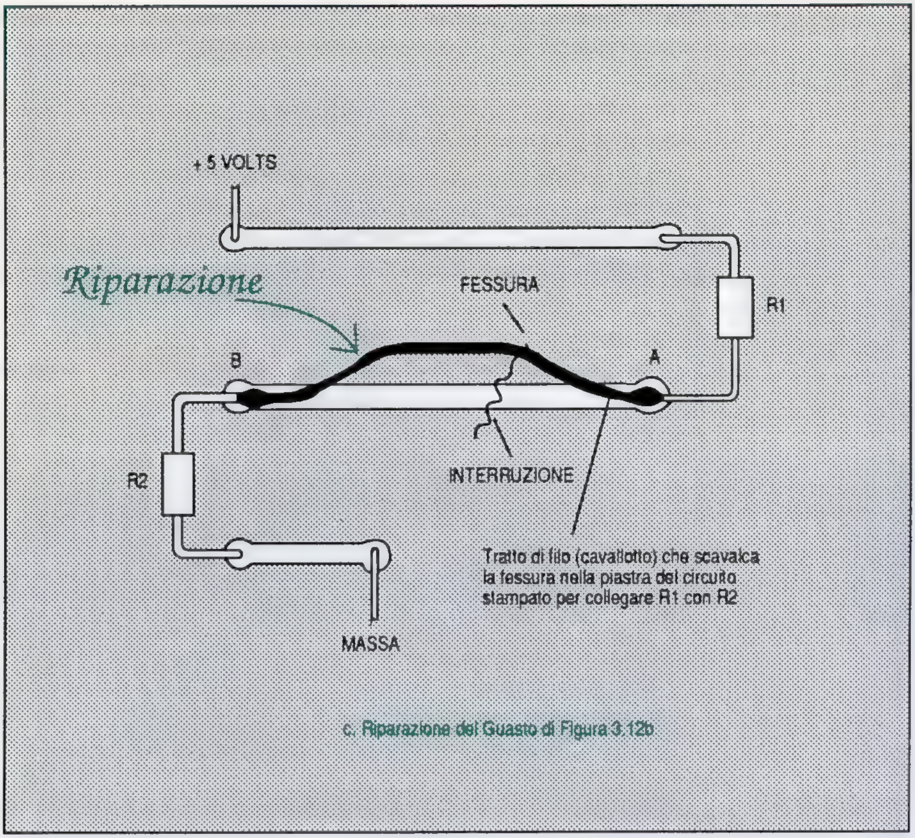
Il corto circuito, mostrato nello schema di *figura 3.1c*, è illustrato con un disegno in *figura 3.12a*. Osservando le caratteristiche del circuito stampato e dei componenti è possibile individuare la presenza di un piccolo pezzo di materiale conduttore, facil-

Figura 3.12. Rivelazione visiva ed elettrica dei guasti fisici (continua)



b. Causa Fisica del Guasto di Figura 3.3a

Figura 3.12. Rivelazione visiva ed elettrica dei guasti fisici
(continua)


+5 VOLTS
Riparazione
FESSURA
R1
B
A
R2
INTERRUZIONE
Tratto di filo (cavallotto) che scavalca la fessura nella piastra del circuito stampato per collegare R1 con R2
MASSA

c. Riparazione del Guasto di Figura 3.12b

mente una goccia di stagno o un pezzettino di filo che, formando un ponte, collega fra loro due rami del circuito stampato. In questo esempio l'effetto della particella conduttrice è quello di mettere in corto la resistenza R1 di *figura 3.12a* e di *figura 3.1c*. Rimuovendo la particella con la sonda di un multimetro o con un qualsiasi altro oggetto di metallo appuntito, come per esempio un cacciavite, il corto può essere rimosso e il circuito riparato. (Non si perda la particella facendola saltar via; essa potrebbe causare un altro corto in un altro punto).

Piccole particelle conduttrici che sono cadute su di un circuito stampato possono mettere in corto circuito le piste conduttrici e causare malfunzionamenti.

Il circuito dovrebbe essere controllato con un voltmetro od un ohmetro prima e dopo che la particella è stata rimossa per confermare che era proprio essa a causare il problema. Non è sempre possibile determinare se una particella è conduttrice semplicemente guardandola, ma anche se la particella fosse senza ombra di dubbio metallica e conduttrice, non sarebbe comunque possibile determinare visivamente se essa sta facendo contatto elettrico con entrambe le piste sul circuito stampato. Comunque, se si misurano 5 volt sia nell'isola A che nell'isola C, come mostrato in *figura 3.12a*, è proprio la particella che sta causando il corto circuito. Dopo che essa è stata tolta la rimozione del corto può essere verificata misurando nell'isola A 2,5 volt rispetto alla massa. Quest'ultima misurazione servirà anche a confermare che R2 non è stata danneggiata dall'eccessiva corrente dovuta al corto ai capi di R1.

Circuito aperto

Una crepa nel conduttore di un circuito stampato è un tipo di apertura che solitamente può essere individuata visivamente. Misurazioni prima e dopo la riparazione assicurano che la crepa è la vera ragione del problema e che è stata riparata.

Nell'esempio di *figura 3.12b* si è verificata un'interruzione che è rappresentata nel circuito di *figura 3.3a*. L'apertura è dovuta ad ùna crepa sul circuito stampato, crepa che corre attraverso il conduttore stampato che collega R1 ad R2. Tali microfratture possono essere difficili da vedere perché anche una microfrattura grande come un capello può interrompere il circuito. Se la presenza di un'apertura non è ancora sicura; a questo punto dovrebbe essere fatta una misura della tensione o della resistenza. Si misuri la resistenza ponendo le sonde di un ohmetro sulle due isole di saldatura circolari A e B con l'alimentatore del sistema spento, o si misuri la tensione ponendo un voltmetro tra l'isola B e la massa come mostrato in *figura 3.12b* con l'alimentatore del sistema acceso. Se la tensione risulta 0 volt nell'isola B e 5 volt nell'isola A, c'è un circuito aperto tra questi due punti e il riparatore può procedere nella riparazione del circuito. La ragione per cui la misurazione è necessaria è che la crepa potrebbe non avere completamente interrotto lo strato di metallo e potrebbe quindi non essere la causa reale del problema verificatosi nel circuito. Una volta confermato che è stata proprio la frattura a causare il circuito aperto, il circuito può essere riparato saldando un filo tra l'isola A e l'isola B come mostrato in *figura 3.12c*. Sarebbe saggio determinare che cosa ha causato la crepa per prevenire danni futuri. Forse è andata persa una vite permettendo così alla piastra di incurvarsi.

L'autore ha incontrato un caso reale in cui si presentava la situazione descritta in *figura 3.12b*: egli si trovò infatti una volta a dover riparare un televisore a colori che non aveva né l'immagine (schermo nero) né il suono. Si trattava di un apparecchio a valvole e l'ispezione visiva indicava che tutte queste funzionavano correttamente. I sintomi (mancanza di audio e schermo nero) facevano sospettare che si trattasse dell'alimentatore a bassa tensione. Un esame visivo di quel circuito individuò invece una crepa, molto simile a quella mostrata in *figura 3.12b*, nel circuito stampato che andava dal collegamento di uscita all'alimentatore. Una misurazione col volt–metro indicava che l'alimentatore produceva 300 volt da un lato della crepa, ma che la tensione dall'altro capo della crepa era di 0 volt. Si spense allora l'apparecchio e fu saldato un filo in modo tale da mettere in collegamento fra loro le due parti del circuito isolate dalla crepa. Immediatamente l'apparecchio riprese a funzionare normalmente.

Naturalmente, semplici corto circuiti e circuiti aperti non sono i soli tipi di guasti che si possono verificare, ma, quando è così, combinare il controllo visivo con le misurazioni elettriche può permettere di individuare chiaramente il guasto. Quando all'interno dei componenti di un sistema si verifica un corto circuito o un circuito aperto o un percorso di dispersione di corrente, molti dei ragionamenti e delle tecniche di misura che vengono utilizzate per trovare corti e aperture nel cablaggio di un sistema possono essere usati per trovare il componente difettoso. Talvolta i componenti difettosi, o almeno il circuito malfunzionante, possono essere localizzati grazie all'apparenza fisica del componente; per esempio, i resistori bruciati.

Guasti nei componenti

I guasti nei circuiti non sono

Se un componente ha un corto circuito o un circuito aperto interno oppure se non ha il corretto livello di tensione di uscita, si deve individuare se il guasto è nel composolo i circuiti aperti, i corto circuiti e le dispersioni di corrente, ma possono essere dovuti anche al malfunzionamento dei componenti.

nente o in qualche parte del circuito esterna al componente. Nel paragrafo seguente saranno trattate diverse tecniche per controllare il funzionamento dei componenti.

Verifica dei componenti tolti dal circuito

Una delle tecniche per controllare un componente richiede la sua rimozione dal sistema e il suo esame con uno strumento adatto al test.

> *ATTENZIONE: prima di rimuovere una parte del sistema si stacchi l'alimentazione e non la si riaccenda finché il pezzo è rimosso, a meno che non si sia sicuri che il sistema non venga danneggiato. Se il sistema è già acceso, lo si spenga prima di inserire la parte.*

Se è possibile rimuovere un componente, un'importante tecnica di ricerca dei guasti consiste nel rimuovere il componente sospetto dal circuito e verificarne il funzionamento esternamente. Una variazione di questa tecnica consiste nel sostituire il componente sospetto con un componente sicuramente funzionante, e di osservare il funzionamento del sistema.

Per esempio, quando si ha a che fare con un televisore a colori che ha delle valvole, e con la misurazione e l'esame visivo non sono stati scoperti nel circuito problemi dovuti a corto circuito o a circuito aperto, allora si dovrebbero rimuovere le valvole dei circuiti sospetti e si dovrebbero controllare con un tester per valvole o sostituire temporaneamente con delle valvole sicuramente funzionanti. Lo stesso procedimento potrebbe essere utilizzato con i transistori e i circuiti integrati se è disponibile una strumentazione adeguata per il test. Se si scopre che un componente è guasto o non perfettamente funzionante, lo si sostituisca. Se, dopo la sostituzione, il sistema non funziona ancora correttamente, si dovrà esaminare il nuovo componente al di fuori del sistema. Se ora è guasto, allora qualcos'altro nel circuito del sistema è guasto ed ha causato il danneggiamento del componente sostituito. Le cause più probabili di questo problema sono un corto sull'uscita del componente, la tensione di alimentazione o le tensioni di ingresso troppo elevate per il componente. In entrambi i casi, l'altro guasto deve essere individuato e riparato per primo. Comunque, nella maggior parte dei casi, la sostituzione di un solo componente guasto consentirà di riparare il sistema.

La rimozione e l'esame esterno di un dispositivo è agevole solo se esso è innestato in uno zoccolo saldato al circuito stampato. Se il componente è saldato direttamente ai collegamenti del circuito stampato, la sua rimozione è molto fastidiosa e può danneggiare il componente o il circuito stampato se la rimozione non è fatta molto attentamente.

In questi casi, il componente dovrebbe essere rimosso solo se si sa per certo che è difettoso; perciò il controllo del funzionamento del componente dovrebbe essere fatto mentre esso è ancora nel sistema. Anche se il componente è su uno zoccolo e può essere rimosso facilmente, può essere possibile isolare il guasto di un sistema riconducendone la causa a quel componente provando il sistema col componente al suo posto. Le tecniche per provare componenti ancora collegati col sistema comportano misurazioni della tensione con e senza tutti i piedini del componente collegati con il sistema. Un procedimento alternativo, detto rilevazione del segnale, implica l'uso combinato di impulsi logici e di sonde o clip logiche per seguire i passaggi dei segnali attraverso vari percorsi nel sistema in modo tale da determinare quali percorsi e quali circuiti sono guasti. Entrambi i metodi verranno discussi qui di seguito.

Verifica di un componente inserito nel circuito

Quando è poco pratico rimuovere un componente dal circuito, i guasti devono essere isolati tramite apposite misurazioni di tensione, di corrente o di resistenza oppure tramite la rilevazione del segnale.

Al fine di esaminare i concetti generali della verifica dei componenti, e specificatamente la verifica dei componenti ancora inseriti nel sistema, verranno considerate di nuovo le situazioni delle *figure 3.2b* e *3.3b*. In *figura 3.2b*, dove il corto circuito si trova da qualche parte sulla linea dalla porta 2 alla porta 3, è relativamente semplice determinare la presenza di un corto; comunque, determinare la posizione precisa in cui si trova il corto è più difficile. Se ciascuna porta è inserita su zoccoli separati, si dovrebbero rimuovere una alla volta e controllare ogni volta con un voltmetro la presenza o meno di un corto circuito. Se, per esempio, si ha un corto circuito quando entrambe le porte 2 e 3 sono ancora nel circuito, ma non più quando la porta 2 viene rimossa, si può dedurre che la porta 2 è guasta e bloccata al livello 0. Se il corto esiste ancora quando la porta 2 è stata rimossa, allora esso deve trovarsi lungo il collegamento fra le due porte o all'ingresso B della porta 3. L'ultimo caso può essere controllato rimuovendo la porta 3 (con o senza la porta 2 reinserita nel circuito). Se la rimozione della porta 3 elimina il corto sulla linea, allora l'ingresso B della porta 3 è in corto con la massa e la porta 3 deve essere sostituita. Se la rimozione di entrambe le porte non elimina il corto circuito significa che è la pista ad aver subito in qualche suo punto il corto a massa, e lo si dovrà eliminare necessariamente. La condizione di circuito aperto di *figura 3.3b* può essere individuata chiaramente con un procedimento simile sebbene le condizioni della pista possano essere controllate facilmente rilevando una misurazione di resistenza tra il terminale di uscita della porta 2 e il terminale di ingresso B della porta 3. Se la resistenza è di 0 ohm la pista non è aperta e il guasto deve trovarsi in una delle porte.

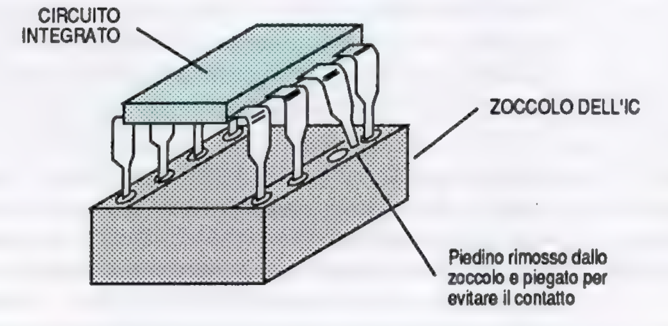
Se le porte delle *figure 3.2b* e *3.3b* non sono su circuiti integrati separati, ma tutte su un solo circuito integrato, non è raccomandabile rimuovere l'intero circuito integrato. In questo caso si potrebbe staccare solo il piedino di uscita della porta 2 e, per la *figura 3.2b* controllare la condizione di corto circuito. (Come identificare i differenti piedini sarà discusso fra breve). Se il corto circuito non viene rimosso compiendo questa azione, dovrà essere staccato il piedino B di ingresso della porta 3. Se una delle due porte risulta guasta si dovrà rimuovere e sostituire l'intero componente.

Come sia possibile staccare un singolo piedino di un componente da un circuito è illustrato in *figura 3.13a* per un circuito integrato inserito in uno zoccolo sulla piastra del sistema e in *figura 3.13b* per un circuito integrato saldato sulla piastra del circuito del sistema. Il caso di *figura 3.13a* è il più semplice in quanto è necessario soltanto rimuovere il circuito integrato dallo zoccolo, curvare leggermente il piedino di uscita in fuori e quindi reinserire il componente nello zoccolo con tutti i piedini del circuito che entrano nello zoccolo eccetto quello curvato. Ora è possibile controllare la porta 2 per cercare il corto circuito in uscita misurando la tensione sul piedino scollegato. Questa tensione non dovrebbe essere bloccata al valore della massa se la porta sta funzionando correttamente. Con due 1 logici in ingresso sulla porta 2 si dovrebbe avere un 1 logico in uscita, secondo la tabella della verità della porta AND. Se il piedino di uscita del dispositivo sembra aver subito un corto a massa indipendentemente dalle condizioni di ingresso, allora la porta 2 è guasta. Se la porta 2 funziona correttamente, si scolleghi il piedino dell'ingresso B della porta 3 e si verifichi il funzionamento della porta 3 con questo piedino di ingresso. Se la porta 3 funziona correttamente, si controlli la resistenza della pista rispetto a massa. Essa dovrebbe essere molto alta o infinita quando sia il piedino di uscita della porta 2 che quello dell'ingresso B della porta 3 sono scollegati dal circuito. Sebbene ciò sia servito per illustrare il metodo di isolare un piedino alla volta, il modo più semplice per trovare questo particolare corto circuito è di rimuovere il circuito integrato e di controllare se esiste il corto tra gli opportuni piedini dello zoccolo e la massa. Se non si manifesta nemmeno un corto circuito, si dovrà sostituire il circuito integrato, perché entrambe le porte sono nello stesso circuito integrato.

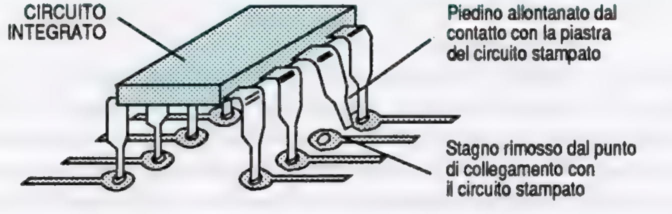
In alcuni casi può essere necessario rimuovere i collegamenti di un singolo piedino del circuito integrato, o un componente discreto, al fine di isolare il guasto del circuito.

La tecnica per isolare i piedini è molto più utile quando il dispositivo è saldato sul circuito stampato, come mostrato in *figura 3.13b*. Le misurazioni vengono effettuate come per l'esempio dell'integrato inserito su uno zoccolo di *figura 3.13a*, ma è molto più difficile staccare il piedino dalla piastra. Generalmente lo si fa tenendo delicatamente il piedino del circuito integrato con delle pinze a punta molto sottile mentre viene applicato del calore per fondere lo stagno dell'isola di saldatura. Quando lo stagno fonde, il piedino viene spinto delicatamente verso l'alto e si elimina il collegamento. Se le misurazioni mostrano che il componente funziona, il procedimento viene ripetuto in senso contrario scaldando l'isola finché lo stagno fonde di nuovo,

Figura 3.13. Tecniche di isolamento di un piedino


CIRCUITO
INTEGRATO
ZOCCOLO DELL'IC
Piedino rimosso dallo zoccolo e piegato per evitare il contatto

a. Isolamento di un Piedino del Circuito nel Caso in cui il Componente sia Inserito su uno Zoccolo


CIRCUITO
INTEGRATO
Piedino allontanato dal contatto con la piastra del circuito stampato
Stagno rimosso dal punto di collegamento con il circuito stampato

b. Isolamento di un Piedino dal Circuito nel Caso in cui il Componente sia Saldato al Circuito Stampato e spingendo delicatamente il piedino nel foro dove si trovava precedentemente. Si potrebbe aggiungere anche del nuovo stagno per completare il collegamento.

Si deve fare molta attenzione nel rimuovere i collegamenti di un singolo piedino di un componente per prevenire il danneggiamento del componente o del circuito.

Nel procedimento di rimozione del piedino è anche possibile (e rende più semplice il lavoro) rimuovere lo stagno dal foro mentre fonde con un dispositivo chiamato dissaldatore. Il tipo più semplice di dissaldatore è simile a una siringa, ma ha la punta resistente al calore. Quando lo stagno è rimosso, dovrebbe essere sufficiente sollevare appena il piedino per staccarlo dalla piastra. Il ricollegamento consiste semplicemente nello spingere il piedino di nuovo nel foro, nel riscaldare sia l'isola che il piedino e nell'applicare del nuovo stagno.

Naturalmente, questo lavoro di saldatura e risaldatura è noioso, lento e potrebbe danneggiare i componenti funzionanti se non viene compiuto con la massima attenzione. Questo metodo dovrebbe essere usato solo come ultimo tentativo per determinare il guasto del sistema. Un metodo più auspicabile e meno distruttivo è quello di seguire i segnali attraverso i percorsi del circuito utilizzando sonde e impulsatori logici. Questa tecnica è conosciuta col nome di rilevazione del segnale.

Rilevazione del segnale

La rilevazione del segnale è una tecnica per individuare i guasti nella quale un segnale noto viene inserito in un sistema in un particolare punto al fine di osservare il funzionamento risultante del sistema per scoprire il malfunzionamento.

La tecnica di rilevazione del segnale, che è stata ampiamente utilizzata nella riparazione di radio e televisori, è utile anche per individuare i guasti nei sistemi digitali. Invece che utilizzare semplicemente un voltmetro o una sonda logica per rilevare la tensione o il livello logico di un dato punto in un sistema in funzione, il metodo di rilevazione del segnale utilizza l'iniezione di un segnale noto in un punto nel sistema e usa quindi gli strumenti di misurazione per determinare come quel segnale si propaga attraverso il sistema.

La tecnica può essere illustrata utilizzando nuovamente il circuito di base di *figura 3.2a*. Il circuito è stato ridisegnato in una forma equivalente contenente solo porte NAND in *figura 3.14a*. Ogni logica composta da un gruppo di porte AND in ingresso che alimentino un gruppo di porte OR in uscita può essere trasformata in una forma che contenga solo porte NAND semplicemente lasciando le interconnessioni così come sono e sostituendo direttamente tutte le porte (AND e OR) con porte NAND. È possibile comperare circuiti integrati contenenti 4 porte NAND a 2 ingressi. Il tipo di diagramma mostrato in *figura 3.14b*, che mostra come queste 4 porte sono collegate con i piedini esterni per un particolare tipo di circuito integrato, viene generalmente fornito dalla casa costruttrice. La *figura 3.14b* mostra anche come i piedini esterni possono essere collegati per ottenere il circuito logico di *figura 3.14a*. Si noti che i numeri che contraddistinguono questi piedini sono identificati nel diagramma logico di *figura 3.14a*. Sulla piastra di un circuito stampato reale i conduttori di collegamento fra i piedini 6 e 9 e tra i piedini 3 e 10 probabilmente correrebbero sotto il circuito integrato.

Rilevazione del segnale tramite sonda logica

La sonda logica e l'impulsatore logico possono essere utilizzati

Il procedimento per la rilevazione del segnale in questo circuito può essere illustrato utilizzando i diagrammi di *figura 3.15*. L'impulsatore è collegato ad uno degli ingressi della porta 1 mentre la sonda è collegata all'uscita del circuito. Durante queper la rilevazione del segnale. La sonda può essere usata come un rilevatore di segnale mentre l'impulsatore come un generatore di segnale.

sta prova, l'ingresso della porta 1 non collegato con l'impulsatore dovrà essere collegato ad un 1 logico per permettere agli impulsi di prova all'altro ingresso di attraversare la porta. Se l'ingresso a cui non è applicato l'impulsatore è fisso sullo 0 logico, il treno di impulsi non può attraversare la porta NAND nemmeno se il circuito funziona correttamente. In questo esempio, perché gli impulsi applicati all'ingresso della porta 1 raggiungano la sonda all'uscita della porta 3, entrambi gli ingressi della porta 2 sono collegati ad uno 0 logico.

Il circuito è simile a quello di *figura 3.2b*, ma con la linea tra la porta 2 e la porta 3 in corto circuito a +5 volt. Generando una sequenza di impulsi sulla linea di ingresso X ed avendo la sonda logica collegata con l'uscita del circuito (piedino 8), la presenza del treno di impulsi viene indicata da una spia lampeggiante sulla sonda logica. Perciò il percorso dall'ingresso X all'uscita funziona correttamente. Analogamente, ponendo l'impulsatore sulla linea Y, collegando la linea X ad un 1 logico e ripetendo la prova, il percorso da Y all'uscita viene verificato. Questa operazione consente di verificare il funzionamento della porta 1 ed un ingresso della porta 3.

Ora, l'impulsatore viene collegato con l'ingresso W (con Z a +5 volt e X ed Y a 0). Non c'è lampeggiamento o luce fissa sulla sonda, ad indicare che il percorso dall'ingresso W all'uscita non sta funzionando. L'impulsatore viene portato all'ingresso Z (con W ora a +5 volt) e si hanno gli stessi risultati. Perciò c'è un guasto su quel percorso. (Se il circuito di *figura 3.3b* (con l'apertura) fosse stato analizzato in questo modo, si sarebbe verificata la stessa sequenza di eventi).

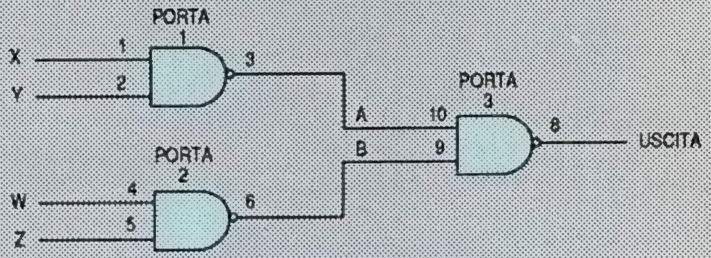
Si porti ora la sonda all'ingresso B della porta 3 (piedino 9). Anche in questo caso non si vede alcuna luce. Ci deve essere un corto circuito da qualche parte tra l'uscita della porta 2, uscita compresa, e l'ingresso B della porta 3, ingresso compreso. Si stacchi il piedino 9 per isolarlo dalla linea. Si colleghi l'impulsatore al piedino 9 e la sonda al piedino 8. Il lampeggiare della spia sulla sonda prova che il corto circuito non è nella porta 3. Ora si scolleghi il piedino 6 per isolare questo dalla linea. Si colleghi la sonda al piedino 6 e l'impulsatore all'ingresso W (piedino 4). La luce lampeggia. Si colleghi l'impulsatore all'ingresso Z (piedino 5). La luce lampeggia verificando così che la porta 2 sta lavorando correttamente. Perciò c'è un corto circuito sul conduttore tra i piedini 6 e 9. Si trovi il corto circuito e lo si rimuova. Questo procedimento viene riassunto nella *tabella 3.1*.

I risultati della rilevazione di un segnale dovrebbero essere analizzati con molta cura. Aperture nel circuito danno risultati differenti rispetto a quelli dei corto circuiti.

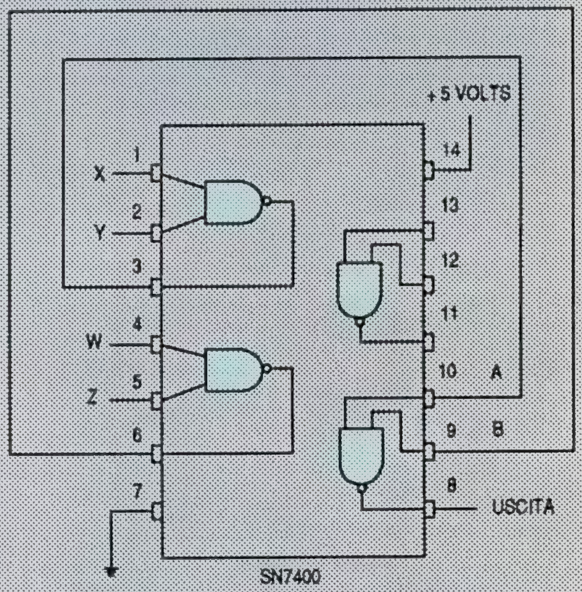
Se, invece di un corto circuito, il problema fosse il circuito aperto di *figura 3.3b*, si otterrebbero risultati differenti dopo aver completato le verifiche della porta 1 collegata con la porta 3. Non ci sarebbe bisogno di staccare alcun piedino. Quando si provasse la porta 2 con l'impulsatore su W, e quindi su Z, questa risulterebbe funzionante come indicato dalla luce lampeggiante nel piedino 6. Poi, il lampeggiamento al piedino 8 con l'impulsatore collegato al piedino 9 proverebbe il funzionamento della porta 3. Ma portando la sonda al piedino 9 con l'impulsatore collegato sia al piedino 4 che al piedino 5, la spia non lampeggerebbe. Ciò indicherebbe che esiste un'apertura nel circuito tra il piedino 6 e il piedino 9. Questo procedimento viene riassunto nella *tabella 3.2*.

Un modo per isolare il punto esatto di interruzione (se si può raggiungere il conduttore) è quello di lasciare l'impulsatore collegato col piedino 4 mentre si sposta la sonda lungo il filo o lungo la metallizzazione del circuito stampato dal piedino 6 verso il piedino 9. L'apertura viene trovata quando la spia smette di lampeggiare.

Figura 3.14. Circuito logico a sole porte NAND

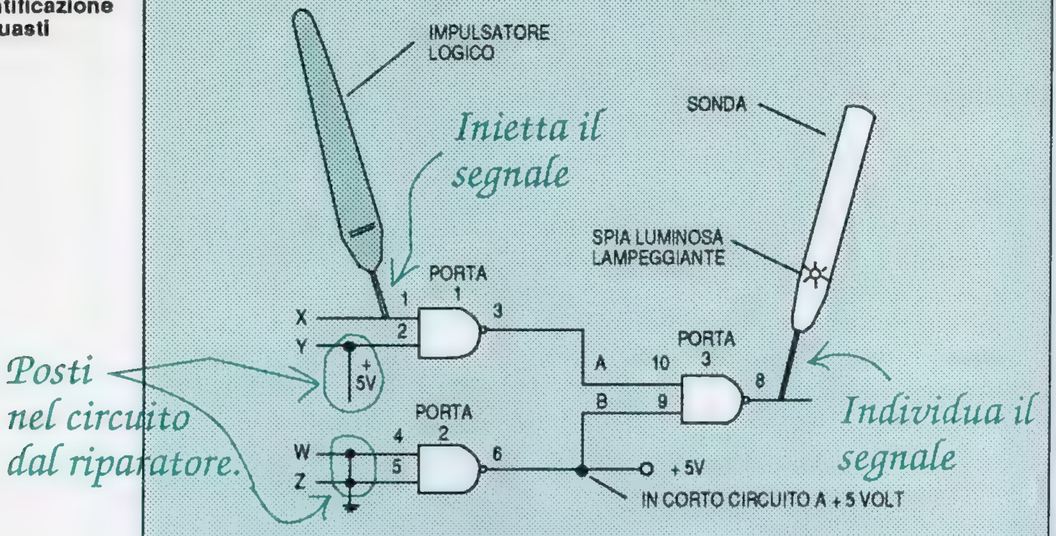

PORTA
1
X
Y
1
2
3
PORTA
3
A
B
10
9
8
USCITA
PORTA
2
W
Z
4
5
6

a. Equivalente del Circuito di Figura 3.2 con sole Porte NAND

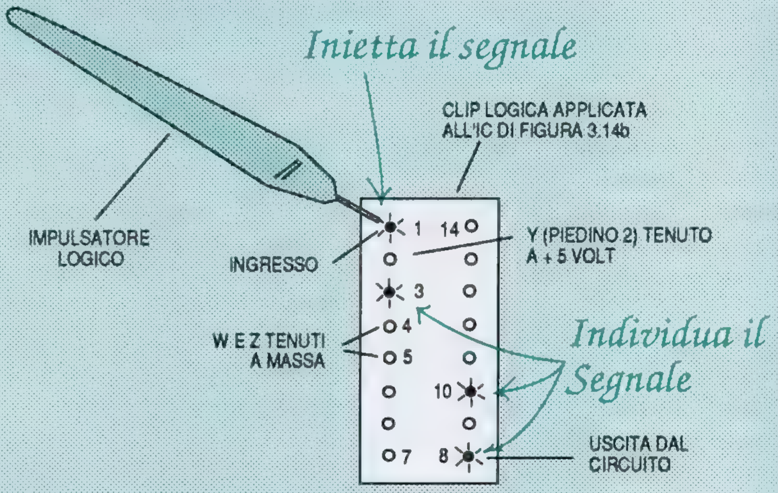

+5 VOLTS
14
1
X
2
Y
3
13
12
11
4
W
5
Z
6
10
A
9
B
7
8
USCITA
SN7400

b. Collegamento Fisico di un Circuito Integrato per il Circuito dell'Esempio a.

Figura 3.15. Rilevazione del segnale per l'identificazione dei guasti

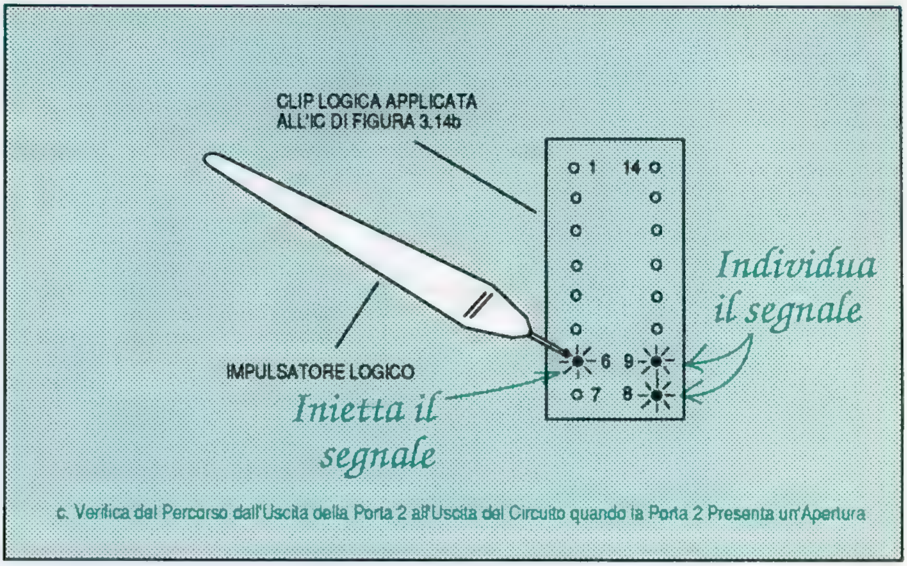

IMPULSATORE LOGICO
Inietta il segnale
SONDA
SPIA LUMINOSA LAMPEGGIANTE
PORTA 1
1
2
3
X
Y
+ 5V
Posti nel circuito dal riparatore.
PORTA 2
4
5
6
W
Z
A
B
10
9
PORTA 3
8
Individua il segnale
+5V
IN CORTO CIRCUITO A +5 VOLT

a. Verifica del Percorso dall'Ingresso X all'Uscita del Circuito Tramite una Sonda Logica


Inietta il segnale
CLIP LOGICA APPLICATA ALL'IC DI FIGURA 3.14b
IMPULSATORE LOGICO
INGRESSO
1
14
Y (PIEDINO 2) TENUTO A + 5 VOLT
3
4
5
W E Z TENUTI A MASSA
10
7
8
Individua il Segnale
USCITA DAL CIRCUITO

b. Verifica del Percorso dall'Ingresso X all'Uscita del Circuito Tramite una Clip Logica

Figura 3.15. Rilevazione del segnale per l'identificazione dei guasti (continua)



c. Verifica del Percorso dall'Uscita della Porta 2 all'Uscita del Circuito quando la Porta 2 Presenta un'Apertura

FIG 3.15 cont

Rilevazione del segnale tramite clip logica

Utilizzare una clip logica piuttosto che una sonda logica velocizza enormemente la rilevazione del segnale in un circuito integrato perché vengono indicati contemporaneamente i livelli di tensione di tutti i piedini del circuito integrato.

L'uso della clip logica insieme all'impulsatore può rendere più veloce l'analisi. In *figura 3.15* la presenza della clip è illustrata da cerchietti che rappresentano i dispositivi emettitori di luce in corrispondenza di ogni singolo piedino del circuito integrato. I cerchietti neri con i raggi indicano che la luce è accesa. Per questo esempio, si assuma che l'uscita della porta 2 sia aperta all'interno del circuito integrato. Si colleghi l'impulsatore all'ingresso X come mostrato in *figura 3.15b*: appariranno delle luci lampeggianti sulla clip logica in corrispondenza dei piedini 1, 3, 8 e 10 come mostrato in figura, per verificare che il percorso del circuito da X all'uscita è funzionante. Ora si colleghi l'impulsatore all'ingresso W: appariranno le spie lampeggianti in corrispondenza dei piedini 2, 3, 8 e 10 a verifica del fatto che il percorso del circuito da Y all'uscita è funzionante.

Portando l'impulsatore all'ingresso W l'unica spia lampeggiante sarà quella in corrispondenza del piedino 4. Portando l'impulsatore all'ingresso Z l'unica spia lampeggiante sarà quella in corrispondenza del piedino 5. Si porti l'impulsatore sul piedino 6: appaiono ora spie lampeggianti in corrispondenza dei piedini 6, 8 e 9 come indicato in *figura 3.15c*. Questo prova che il percorso dal piedino 6 all'uscita del circuito funziona; perciò deve essere difettosa la porta 2.

Il modo migliore per compiere la riparazione è quello di sostituire il circuito integrato. Se il circuito si trovasse in un pannello sperimentale di sviluppo nel quale per i collegamenti si siano usati fili conduttori, la porta guasta potrebbe essere scollegata dal circuito e la quarta porta inutilizzata potrebbe essere collegata al suo posto.

I vantaggi del metodo di rilevazione del segnale sono che esso è veloce e semplice e permette di controllare il funzionamento di ciascuna parte del circuito sospettato senza rimuovere il circuito o uno qualsiasi dei suoi componenti. La rilevazione del segnale insieme alle misurazioni di tensione e resistenza permetterà al riparatore di individuare e riparare la maggior parte dei guasti più semplici e comuni che si presentano nei sistemi elettronici digitali e non.

Tabella 3.1. Rilevazione del segnale nel circuito di figura 3.14a con un corto circuito a +5 volt causato dal conduttore tra i piedini 6 e 9

Piedino dell'IC cui è Collegato l'Inpulsatore logico	Piedino dell'IC cui è Collegata la Sonda Logica	Indicazione della Sonda	Conclusione
1	8	Lampeggia	
2	8	Lampeggia	Porta 1 OK
4	8	Spenta	
5	8	Spenta	
9	8	Spenta	
Isolato 9	8	Lampeggia	Porta 3 OK
4	Isolato 6	Lampeggia	
5	Isolato 6	Lampeggia	Porta 2 OK Corto circuito esterno a +5 volt

Tabella 3.2. Rilevazione del segnale nel circuito di figura 3.14a con un'apertura nel conduttore tra i piedini 6 e 9

Piedino dell'IC collegato all'Impulsore Logico	Piedino dell'IC collegato alla sonda logica	Indicazione della Sonda	Conclusione
1	8	Lampeggia	
2	8	Lampeggia	Porta 1 OK
4	8	Spenta	
5	8	Spenta	
4	6	Lampeggia	
5	6	Lampeggia	Porta 2 OK
9	8	Lampeggia	Porta 3 OK
4	9	Spenta	
5	9	Spenta	Apertura Esterna tra i Piedini 6 e 9

CHE COSA ABBIAMO IMPARATO?

1. I guasti più comuni che si verificano in un qualsiasi sistema digitale sono i corto circuiti, i circuiti aperti e i problemi dei livelli di tensione.
2. Questi guasti possono essere individuati facilmente eseguendo misurazioni adeguate di tensione e resistenza in ogni parte sospetta del sistema.
3. Queste misurazioni possono essere effettuate sia con uno strumento analogico che con uno strumento digitale o una sonda logica.
4. Il riparatore deve osservare il funzionamento del sistema e utilizzare le sue conoscenze sulle funzioni del sistema per determinare qual è la funzione che più probabilmente ha causato il problema osservato.

5. Una volta localizzato il problema in una data unità funzionale del sistema, per localizzare ed eliminare il guasto si potranno usare le tecniche del controllo visivo, delle misurazioni della tensione e delle misurazioni effettuate con la sonda, e della rilevazione del segnale insieme alla verifica esterna del funzionamento di un componente o alla sostituzione del componente.

Quiz del Capitolo 3

1. Se nel circuito di *figura 3.1a* il valore di R2 viene aumentato fino a 9000 ohm, quale sarebbe la tensione su R2?
   a. 0
   b. 0,5
   c. 4,5
   d. 5,0

2. Se nel circuito di *figura 3.1b* R2 sviluppasse un circuito aperto interno (non passasse cioè corrente attraverso R2) quale sarebbe la corrente nel circuito in milliampere (la corrente di alimentazione)?
   a. 0
   b. 2,5
   c. 5,0

3. Se il guasto descritto nel problema 2 si verificasse nel circuito di *figura 3.1c*, quale sarebbe la corrente dell'alimentazione in milliampere?
   a. 0
   b. 2,5
   c. 5,0

4. Se nel circuito di *figura 3.2a* si verificasse un corto circuito tra l'uscita della porta 1 e l'alimentatore che fornisce una tensione di 5 volt, da che cosa dipenderebbe l'uscita del circuito?
   a. Dall'ingresso X
   b. Dall'ingresso Y
   c. Dagli ingressi W e Z
   d. Nessuno dei casi precedenti

5. Se nel circuito di *figura 3.2a* si verificasse un corto circuito tra una linea dell'alimentatore e l'uscita della porta 2, che livello logico si avrebbe in uscita?
   a. 0
   b. 1
   c. 0 o 1 a seconda dei valori di X e Y

6. Se nel circuito di *figura 3.3a* si verificasse un circuito aperto all'interno di R2 invece che nel filo, quale sarebbe la corrente di alimentazione in milliampere?
   a. 0
   b. 2,5
   c. 5,0

7. Sotto le stesse condizioni della domanda 6, quale sarebbe la tensione ai capi di R2?
   a. 0
   b. 2,5
   c. 5,0

8. In *figura 3.4a* la tensione in uscita è approssimata a 0 volt essendo la corrente di base di 50 microampere. Se si considera la caduta di tensione di 0,6 volt tra la base e l'emettitore, quale sarebbe il valore della corrente di collettore in microampere per una tensione di ingresso di 5 volt?
   a. 3,0
   b. 4,0
   c. 4,4
   d. 4,7

9. Sotto le stesse condizioni della domanda 8, quale sarebbe la tensione di uscita?
   a. 2,0
   b. 1,0
   c. 0,8
   d. 0,6
   e. 0,2

10. Nel circuito di *figura 3.4d* quale sarebbe la corrente di collettore in milliampere se la resistenza di 40 kiloohm fosse portata a 100

kiloohm supponendo che la tensione in ingresso sia di 5,6 volt?
a. 3,0
b. 4,0
c. 4,4
d. 4,7

11. Quale sarebbe la tensione di uscita nel circuito di *figura 3.4d* se si applicasse una tensione di ingresso di 5 volt?
a. 2,1
b. 2,7
c. 1,6
d. 0,8

12. Se nel circuito di *figura 3.4d* il valore di R_e fosse diminuito a 500 ohm, quale sarebbe la tensione di uscita?
a. 0
b. 1,5
c. 3,25
d. 4,0

13. Se si collegasse una resistenza di 50 kiloohm in parallelo con una resistenza di 25 kiloohm, quale sarebbe la resistenza equivalente risultante dal collegamento?
a. 25 kiloohm
b. 20 kiloohm
c. 16,67 kiloohm
d. 16 kiloohm

14. Se nel circuito di *figura 3.6a* lo strumento avesse una resistenza di ingresso di 100 kiloohm, quale sarebbe la tensione misurata ai capi di R2?
a. 1,78
b. 2,38
c. 2,43
d. 2,50

15. Mentre si effettua la rilevazione di un segnale si ponga una sonda logica sull'uscita della porta 3 di *figura 3.2a*. Quando viene applicato un impulsatore ad entrambi gli ingressi della porta 3, sulla sonda si ottiene un lampeggiamento. Se l'impulsatore viene portato all'uscita della porta 1, il lampeggiamento non si verifica. Qual è il guasto nel circuito?
a. C'è un corto circuito tra la porta 1 e l'alimentazione di +5
b. C'è un corto circuito nel filo che collega la porta 3 con la porta 2
c. C'è un'apertura nel filo che collega la porta 1 con la porta 3
d. C'è un'apertura nel filo che collega la porta 2 con la porta 3

16. Se nel circuito di *figura 3.2a* collegando l'impulsatore a ciascuno degli ingressi W, Y e Z si ottiene un lampeggiamento sulla sonda collegata con l'uscita della porta 3, ma collegandolo con X questo non accade, qual è il guasto nel circuito?
a. È guasta la porta 1
b. La linea di ingresso tra X e la porta 1 è aperta
c. È guasta la porta 3
d. Sono vere sia a che b.

17. Se nel circuito di *figura 3.2a* collegando l'impulsatore a un qualsiasi ingresso X, Y, W o Z si ottiene un lampeggiamento sulla sonda su qualunque ingresso della porta 3, qual è il problema?
a. È guasta la porta 1.
b. È guasta la porta 2
c. I due ingressi della porta 3 sono in corto fra loro

18. Se in un televisore a colori si ha l'audio ma non il video, ma lo schermo è illuminato, qual è l'elemento funzionale che più probabilmente si è guastato?
a. L'amplificatore RF
b. Il sintonizzatore
c. Il rilevatore video
d. I circuiti del colore
e. Gli amplificatori di deflessione

19. Nel circuito di *figura 3.1a* viene misurata una caduta di tensione di 0 volt ai capi di R2 e la resistenza misurata tra il lato sinistro di R1 e la massa è di 2000 ohm con l'alimentazione staccata. Che cosa non va nel circuito?
a. C'è un corto circuito ai capi di R2
b. C'è un'apertura in R2
c. C'è un corto circuito ai capi di R1
d. L'alimentatore è guasto

20. Se nel circuito di *figura 3.1a* viene misurata una tensione di 0 volt ai capi di R2, quali sono i guasti possibili?
a. Un circuito aperto
b. R2 in corto circuito
c. Un'alimentazione errata
d. Un corto ai capi dell'alimentatore
e. Tutte le risposte precedenti
f. Le risposte a e b precedenti

ASPETTI DI LOGICA COMBINATORIA

SCOPO DEL CAPITOLO

Nel capitolo 3 sono state introdotte le cause elettriche della maggior parte dei guasti che si riscontrano nei sistemi elettrici ed elettronici insieme con le tecniche di misurazione che possono essere utilizzate per localizzare la posizione di tali guasti. Sono stati introdotti anche alcuni dei guasti che si verificano nei circuiti logici combinatori. Questo capitolo continua l'esame di tali guasti e fornisce alcuni metodi che possono essere utilizzati per riparare i sistemi logici combinatori. Scopo principale di questo capitolo sarà quello di far comprendere il funzionamento dei sistemi e dei componenti, essendo queste conoscenze necessarie per trovare un guasto.

GUASTI NEI CIRCUITI DIGITALI SSI

Semplici combinazioni di porte vengono utilizzate anche nei sistemi più complessi per controllare il funzionamento dei sottosistemi e le funzioni d'ingresso/uscita.

Le fondamentali porte utilizzate nei moderni sistemi digitali vengono fabbricate in circuiti integrati a piccola scala di integrazione (SSI) con meno di 100 porte per chip. Le caratteristiche operative e i simboli logici per le porte AND, OR, NOT, NAND e NOR sono stati trattati nei capitoli precedenti. Se il lettore non ricordasse perfettamente come funzionano queste porte, torni indietro e rilegga la parte che ne tratta perché queste conoscenze sono fondamentali e necessarie per la comprensione di quanto segue.

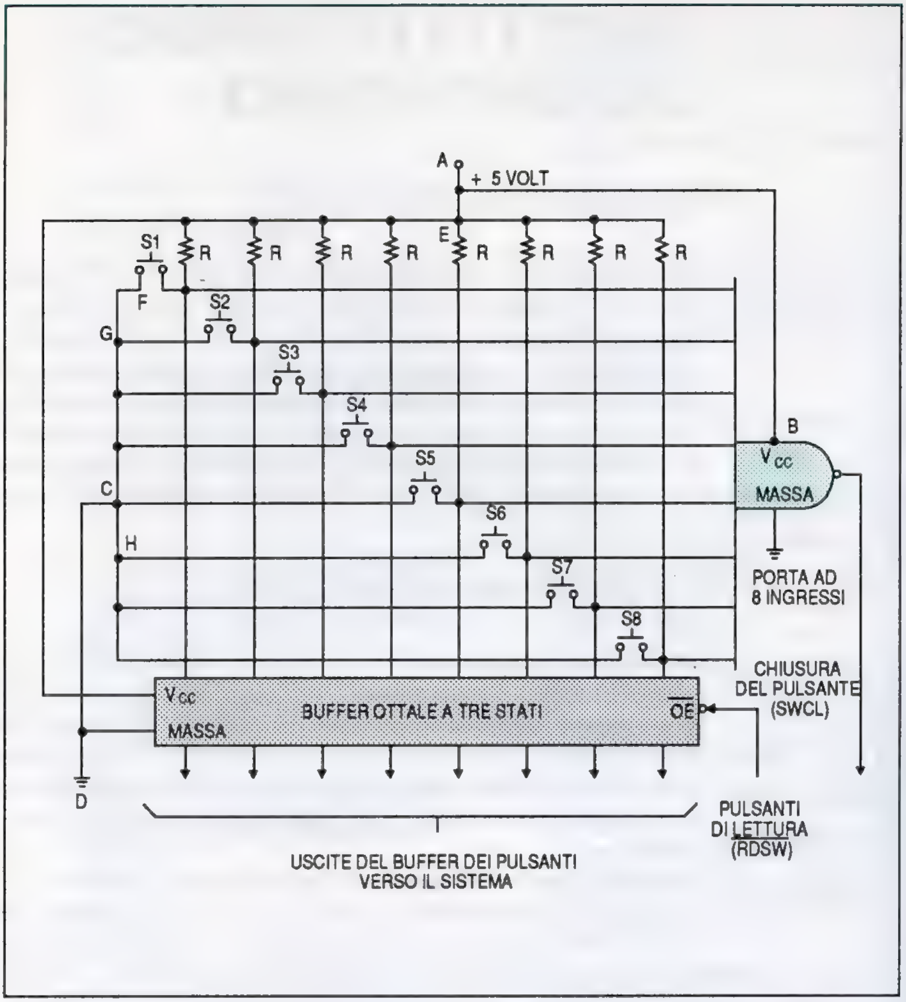
I sistemi tipici sono formati da un gruppo di porte NAND in ingresso collegato con un gruppo di porte NAND in uscita, oppure sono una struttura simile formata da sole porte NOR. Anche i sistemi relativamente complessi come i computer utilizzano alcune di queste porte per controllare il funzionamento e le funzioni di ingresso/uscita dei componenti del sistema. In questi casi la logica combinatoria può consistere di una sola porta NAND cui è associato un componente integrato su media o grande scala. (Un componente integrato su media scala (MSI) ha più di 100 porte per chip e un componente integrato su larga scala (LSI da Large-Scale Integration) ha più di 1000 porte per chip). Un semplice esempio di una situazione simile è illustrato in *figura 4.1*: si tratta del circuito di una tastiera ad 8 interruttori. Il circuito è composto da 8 resistori, da una porta NAND a 8 ingressi, da 8 pulsanti e da un buffer ottale a tre stati (3-state).

Funzionamento del buffer a tre stati

I buffer a tre stati, controllati da un segnale di abilitazione comune.

Per capire il funzionamento del circuito di *figura 4.1* è necessario ricordare il funzionamento del buffer a tre stati. (Si ricordi che questo funzionamento è stato descritto nel capitolo 2.) Il diagramma logico per un buffer ottale a tre stati, mostrato in *figura 4.2a*, consiste di 8 (da ciò il nome ottale) buffer, uno per ogni singolo bit, con

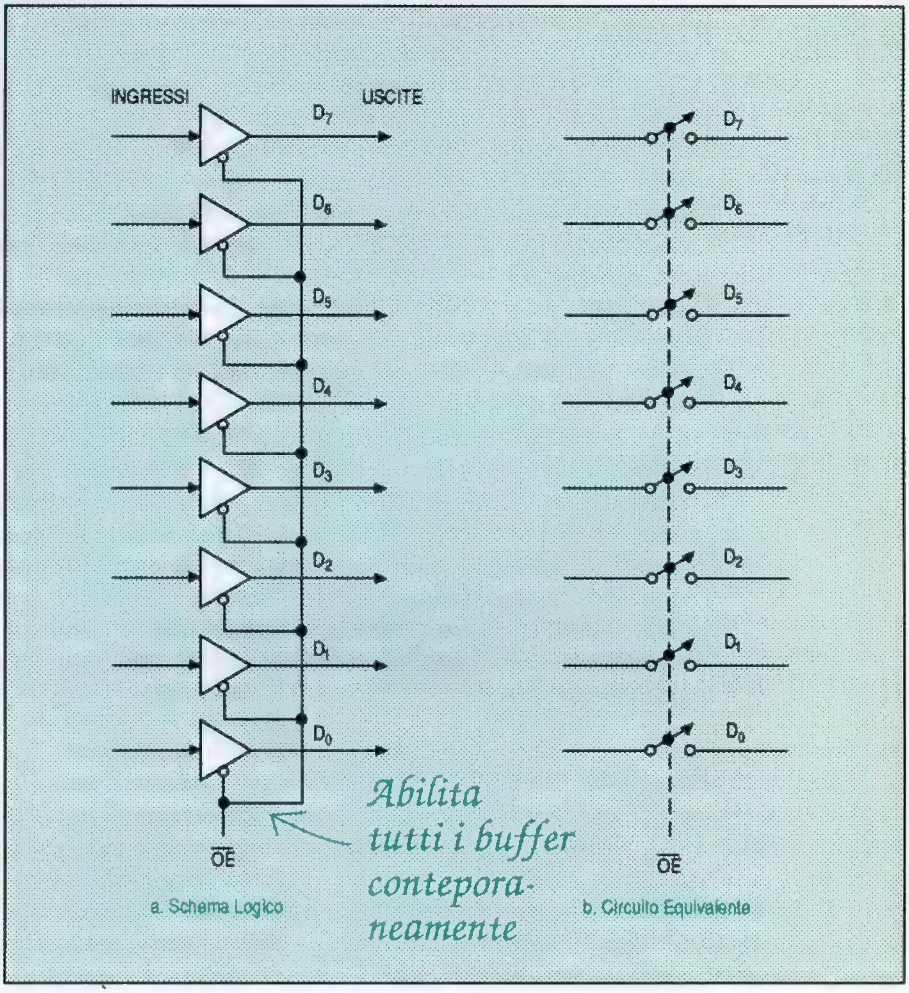
Figura 4.1. Logica combinatoria che utilizza semplici porte


A
+ 5 VOLT
E
S1
S2
S3
S4
S5
S6
S7
S8
R
F
G
C
H
B
V_{CC}
MASSA
PORTA AD 8 INGRESSI
CHIUSURA DEL PULSANTE (SWCL)
BUFFER OTTALE A TRE STATI
$\overline{OE}$
D
PULSANTI DI LETTURA (RDSW)
USCITE DEL BUFFER DEI PULSANTI VERSO IL SISTEMA

vengono utilizzati per trasferire bit in parallelo su un bus comune nei microcomputer e in altri sistemi digitali.

le linee $\overline{OE}$ (output enable not) collegate fra loro in parallelo. I segnali di ingresso ai livelli logici 0 ed 1 appariranno sulle linee di uscita solo quando il segnale di controllo $\overline{OE}$ è al suo livello attivo, cioè se è al livello basso (livello 0). Si ricordi che ciò viene indicato dal circolino sugli ingressi di controllo dei buffer e dalla barretta sopra il nome del segnale. Perciò solo quando la linea di controllo OE è portata al livello logico 0 le linee di uscita riceveranno i livelli di tensione di ingresso. D'altra parte, quando la linea $\overline{OE}$ è al livello logico 1, le linee di uscita sono elettricamente scollegate dalle linee di ingresso e appariranno essenzialmente come circuiti aperti. Per questo i buffer possono essere pensati come interruttori elettrici la cui chiusura viene controllata dal segnale $\overline{OE}$, come illustrato in *figura 4.2b*. Il buffer a tre stati viene usato comunemente nei microcomputer per controllare i collegamenti dei diversi circuiti con un bus comune.

Figura 4.2. Funzionamento di un buffer ottale a tre stati


INGRESSI
USCITE
D_7
D_6
D_5
D_4
D_3
D_2
D_1
D_0
$\overline{OE}$
Abilita tutti i buffer conteporaneamente
a. Schema Logico
b. Circuito Equivalente

Funzionamento del circuito

Ora che è stato capito il principio di funzionamento del buffer a tre stati, può essere analizzato il funzionamento del circuito di *figura 4.1* per fornire le basi necessarie per il ragionamento sui guasti che possono verificarsi in un circuito di questo tipo. In *figura 4.1* ci sono 8 pulsanti; un terminale di ciascun pulsante è collegato a massa (0 volt). L'altro terminale è collegato con un resistore, con uno degli ingressi della porta NAND e con uno degli ingressi del buffer ottale a tre stati. Poiché l'altro capo del resistore è collegato con +5 volt, alla porta NAND e al buffer ottale a tre stati è applicato un 1 logico quando i contatti del pulsante sono aperti, come mostrato in figura. Se tutti i pulsanti sono aperti, tutti gli ingressi al buffer e alla porta NAND sono a livello logico 1 e quindi l'uscita della porta NAND è a livello logico 0. Quan-

Utilizzando una combinazione di porte NAND e un buffer ottale a tre stati, gli ingressi provenienti da una tastiera ad 8 tasti vengono interfacciati con un microcomputer.

do viene premuto il pulsante, i contatti si chiudono per collegare l'ingresso della porta NAND e il buffer ottale a tre stati con la massa. Questo fa sì che sull'ingresso della porta NAND ci sia uno 0 logico, il che a sua volta costringe l'uscita della porta al livello logico 1. Perciò, se uno o più pulsanti sono chiusi, l'uscita della porta NAND è al livello 1 e questa uscita, chiamata SWCL (Switch closure, pulsante chiuso), indica al circuito successivo che c'è almeno un pulsante chiuso. Per ogni linea che ha un pulsante chiuso, viene posto uno 0 logico anche all'ingresso corrispondente del buffer ottale a tre stati. Perciò alcuni ingressi al buffer possono essere 0 ed altri possono essere 1.

Si supponga che il circuito di *figura 4.1* sia una tastiera ad 8 pulsanti collegata con un microcomputer. Quando viene premuto un qualsiasi tasto, SWCL passa ad un 1 logico e ciò indica al microcomputer che un tasto è stato premuto. Il microcomputer può determinare qual è il tasto che è stato premuto leggendo il codice a 8 bit nel buffer, portando l'ingresso $\overline{\text{OE}}$ del buffer a tre stati ad un valore basso. Il segnale proveniente dal microcomputer che svolge questa operazione viene chiamato $\overline{\text{RDSW}}$ (da Read Switches Not, lettura dei pulsanti). Perciò il microcomputer sa che c'è un tasto premuto per ogni bit a livello logico 0. Espandendo il circuito di *figura 4.1* in modo da includere più pulsanti, è possibile interfacciare una tastiera complessa, come quella di una macchina per scrivere, a un microcomputer o ad un altro sistema digitale. Poiché una tastiera è un dispositivo di ingresso comune in questi sistemi, il riparatore di sistemi digitali si ritroverà probabilmente ad affrontare guasti in questo tipo di circuiti.

Ricerca ed eliminazione dei guasti

Il circuito di *figura 4.1* è diretto e relativamente semplice, cosicché i tipi di guasti ed i loro effetti possono essere determinati abbastanza facilmente. Esamineremo ora i guasti più probabili.

SWCL bloccato a 0

Se nell'esempio della tastiera il segnale di controllo SWCL rimanesse bloccato a 0 i punti da controllare sarebbero: un guasto nell'alimentazione a +5volt, un possibile guasto in una porta NAND, o il guasto di una linea comune a tutti gli otto pulsanti.

Il caso di guasto più semplice da determinare è quello che si ha quando SWCL rimane bloccato sullo 0 logico quando viene chiuso un qualsiasi pulsante. Il microcomputer non verrebbe mai informato di questa chiusura, perciò non cercherebbe mai di leggere il codice binario dal buffer. L'unica condizione per cui SWCL può rimanere a 0 si verifica solo se tutti gli 8 ingressi della porta NAND sono bloccati al livello 1 o se la stessa uscita della porta NAND è bloccata al livello 0. Con buona probabilità questo guasto è causato da una mancanza dei +5 volt di alimentazione alla porta NAND, da un guasto alla porta stessa, o da qualche linea comune a tutti gli 8 pulsanti.

Si controlli se ci sono +5 volt nel punto B (l'ingresso V_{cc} alla porta NAND). Se non ci sono, si controlli nel punto A. Se ci sono +5 volt nel punto A ma non nel punto B, si provi a cercare lungo il filo di collegamento tra i due punti per trovare il circuito aperto.

Se invece ci sono +5 volt nel punto B, si può controllare la porta NAND nel circuito utilizzando una sonda logica ed un impulsatore per determinare se è possibile por-

La sonda logica e l'impulsatore vengono utilizzati per determinare se la porta NAND è guasta o se il problema si trova nella matrice dei pulsanti.

tare SWCL ad un livello alto quando si abbassa il livello dell'ingresso. Perciò, applicando un impulso su una qualsiasi delle linee di ingresso e collegando la sonda sulla linea SWCL si dovrebbe avere il lampeggiamento della spia sulla sonda se la porta sta lavorando correttamente. Se ciò non avviene, la porta deve essere sostituita. Se la porta sembra rispondere correttamente a tutte le linee di ingresso, allora il guasto si trova in qualche altra parte del circuito.

Se applicando impulsi alla porta NAND si vede che essa sta funzionando bene, si dovrà controllare la matrice di collegamento dei pulsanti. L'unica parte del circuito comune a tutti i pulsanti è il filo dal punto C al punto D che li collega a massa. Se questo collegamento comune è aperto, nessuno dei pulsanti può fornire uno 0 logico agli ingressi della porta NAND. Controllare questo percorso con un ohmetro dovrebbe servire a localizzare l'apertura che, molto probabilmente, si sarà verificata nei collegamenti nei punti C o D, ma che potrebbe verificarsi in un qualsiasi punto lungo questo percorso. Naturalmente, il circuito potrà essere riparato installando un filo che ripristini il collegamento interrotto dall'apertura.

Segnale SWCL bloccato a 1

Quando il segnale di controllo è bloccato a 1 tramite il buffer viene indicata quale delle tre possibili situazioni si è verificata: un pulsante chiuso, tutti i pulsanti chiusi bloccati, oppure nessun pulsante chiuso.

Un altro sintomo di guasto si ha quando SWCL è bloccato a 1. In questo caso, il microcomputer leggerebbe continuamente il buffer solo per trovare che nessun pulsante è stato chiuso, oppure che un pulsante è sempre chiuso o che tutti sembrano essere chiusi.

Il primo caso da considerare è quello in cui SWCL è 1 anche se tutti i pulsanti sono aperti. Il problema più probabile è che la porta NAND sia guasta oppure che ci sia un'apertura sul collegamento a massa della porta NAND. Entrambi questi problemi impedirebbero alla porta NAND di generare uno 0 logico anche quando tutti gli ingressi della porta sono al livello 1.

Per quanto riguarda il caso nel quale sembra che un pulsante sia sempre chiuso, ci potrebbe essere un circuito aperto fra il terminale del pulsante e il punto E. Questo percorso comprende il resistore che potrebbe facilmente essere la causa del problema. Un'altra causa potrebbe essere un corto a massa sulla linea che collega quel pulsante con la porta NAND o sulla linea che lo collega con il buffer. Questo corto circuito potrebbe essere nel pulsante, nel circuito di ingresso della porta NAND, nel circuito di ingresso del buffer o nel cablaggio. L'isolamento e la prova dei componenti, come descritti nel capitolo 3, permetteranno di localizzare il corto a massa. Un altro caso che resta da considerare quando SWCL è bloccato ad 1 è quello in cui tutti i pulsanti sembrano permanentemente chiusi (tutte le uscite del buffer a 0). Ciò può essere probabilmente causato da un circuito aperto tra il punto A di collegamento dell'alimentatore e il punto E comune ai resistori, con il cammino da A a B intatto. Si può escludere un corto sul cammino da A a B perché SWCL non potrebbe essere 1 se la linea a +5 volt avesse subito un corto circuito.

Funzionamento non corretto di un solo interruttore

Considereremo ora un ultimo guasto possibile nel circuito di *figura 4.1*. Si supponga che il circuito risponda normalmente a tutte le chiusure dei pulsanti tranne che a quelle di S1. Questo problema viene individuato dal sintomo: o il pulsante è guasto e non fa contatto, o c'è un'apertura nel cammino dal pulsante (punto F) a massa (punto G), o c'è un'apertura tra l'interruttore e l'ingresso della porta NAND, o quel particolare ingresso della porta NAND è guasto. Misurazioni di resistenza e controlli con la sonda logica dovrebbero isolare il problema. Se il pulsante è guasto, dovrà essere sostituito. Comunque, in una emergenza, se i contatti possono essere raggiunti, strofinare i punti di contatto con carta vetrata o tela abrasiva potrebbe portare il pulsante a funzionare ancora correttamente.

Anche se semplici circuiti combinatori di porte possono avere diverse fonti di guasto, analizzare il funzionamento del circuito e utilizzare una semplice strumentazione e procedura per il test servirà ad isolare il guasto.

Ci sono altri guasti che potrebbero verificarsi nel circuito di *figura 4.1* che non sono stati discussi. Per esempio, il buffer potrebbe non rispondere al segnale $\overline{\text{RDSW}}$ o potrebbe far transitare il segnale da uno degli ingressi all'uscita in maniera non affidabile. Comunque, questi ed altri guasti in questo tipo di circuiti possono essere isolati utilizzando ragionamenti e procedimenti simili a quelli dati negli esempi. Questi esempi hanno illustrato che anche in un semplice circuito combinatorio a porte logiche possono esserci molte cause diverse di guasto. Comunque, esaminando il funzionamento del circuito razionalmente, il guasto solitamente può essere isolato utilizzando semplici strumentazioni e procedimenti. Questo stesso tipo di approccio funzionerà anche nei più complessi circuiti integrati a media e larga scala (MSI e LSI) che verranno trattati nella restante parte di questo capitolo.

GUASTI NEI CIRCUITI DIGITALI MSI

I circuiti MSI hanno meno collegamenti e presentano una frequenza più bassa di guasti. Ne risulta un sottosistema in cui i guasti possono essere isolati in maniera relativamente semplice.

I circuiti MSI contengono circuiti come decodificatori, multiplexer, contatori, registri a scorrimento e codificatori. Questi dispositivi vengono usati insieme a semplici porte per svolgere funzioni digitali relativamente complicate con solo uno o due circuiti integrati. Ne risulta un sottosistema con meno interconnesioni e minori possibilità di guasti. Quando si verifica un guasto è relativamente facile isolarlo in quanto esso deve trovarsi in uno dei pochi circuiti o nei pochi fili utilizzati per collegare fra loro i circuiti. In questa parte del capitolo verranno considerati il funzionamento e i guasti di alcuni dei componenti MSI più comuni.

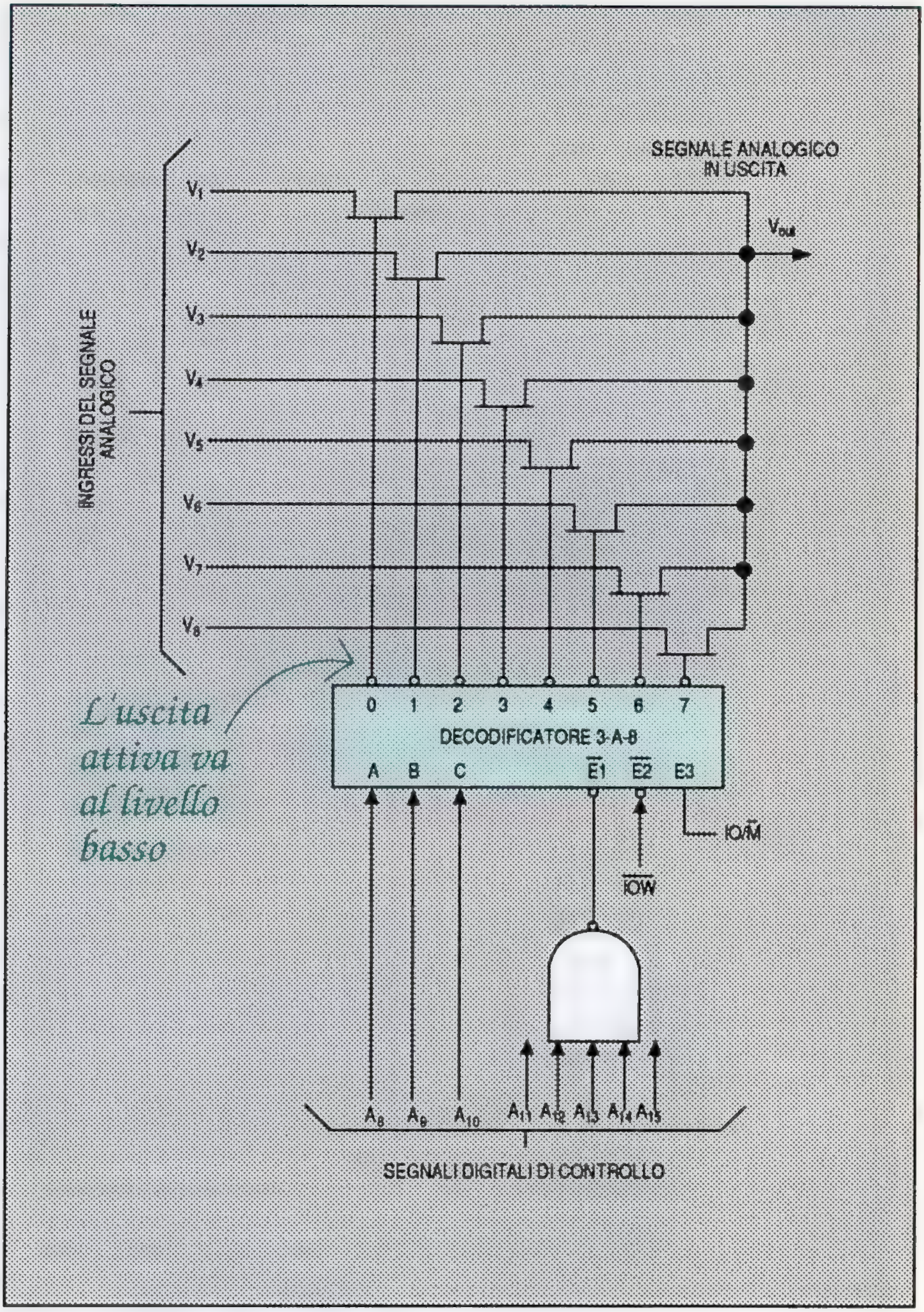
Multiplexer analogico

Il circuito MSI più comunemente usato è il decodificatore o multiplexer. Un'applicazione che utilizza il più semplice decodificatore logico digitale con transistori discreti MOSFET per svolgere la funzione di multiplexer analogico è mostrata in *figura 4.3*. I segnali di controllo digitali, le linee tra A_8 e A_{15}, il segnale $\overline{\text{IOW}}$ e il segnale IO/$\overline{\text{M}}$, potrebbero provenire da un microcomputer o da un altro sistema di controllo digitale.

Il decodificatore è il circuito MSI più comune ed è progettato in modo tale che solo uno alla volta dei suoi segnali d'uscita

I segnali che controllano il funzionamento del circuito vengono indicati con nomi abbastanza comuni se sono generati da un microprocessore. Le linee da A_8 a A_{15} vengono chiamate linee di indirizzo perché esse selezionano o indirizzano una data linea di uscita del decodificatore. I segnali $\overline{\text{IOW}}$ e IO/$\overline{\text{M}}$ sono segnali di temporizzazione e di controllo che determinano quando il multiplexer analogico è abilitato o attivato. Il segnale $\overline{\text{IOW}}$ (Input Output Write Not) diventa basso quando il processo-

Figura 4.3. Multiplexer 8 a 1 per segnali analogici


SEGNALE ANALOGICO IN USCITA
V_{out}
INGRESSI DEL SEGNALE ANALOGICO
V_1
V_2
V_3
V_4
V_5
V_6
V_7
V_8
0 1 2 3 4 5 6 7
DECODIFICATORE 3-A-8
A B C
$\overline{E1}$ $\overline{E2}$ E3
$IO/\overline{M}$
$\overline{IOW}$
A_8 A_9 A_{10} A_{11} A_{12} A_{13} A_{14} A_{15}
SEGNALI DIGITALI DI CONTROLLO

L'uscita attiva va al livello basso

possa essere al livello basso.

re vuole inviare informazioni al dispositivo I/O prescelto o vuole attivare quest'ultimo. Il segnale IO/$\overline{\text{M}}$ viene utilizzato dal processore per indicare se deve essere abilitato il sottosistema di ingresso-uscita o il sottosistema di memoria. Se il segnale è a 0 viene abilitato il sottosistema di memoria; se è ad 1 viene abilitato quello di ingresso-uscita.

Gli ingressi A, B, C del decodificatore sono noti come linee di selezione del decodificatore in quanto i segnali sulle linee A_8, A_9 e A_{10} selezionano l'uscita del decodificatore. Le linee dalla A_{11} alla A_{15} sono collegate con una porta NAND la cui uscita sarà al livello logico 0 solo se tutte e cinque queste linee di segnale sono al livello logico 1. L'uscita della porta NAND così come i segnali di controllo $\overline{\text{IOW}}$ e IO/$\overline{\text{M}}$ sono collegati con gli ingressi di attivazione $\overline{\text{E1}}$, $\overline{\text{E2}}$ e E3. Le otto uscite attive basse del decodificatore sono numerate da 0 a 7. Queste uscite sono collegate con i gate di otto transistori MOSFET per controllare quando devono entrare in conduzione. (L'ingresso di controllo di un FET è detto gate invece che base).

Quando è applicato un livello logico 0 al gate, il transistore agisce come un resistore di valore ohmico molto basso, cosicché la tensione dell'ingresso analogico è collegata con la linea di uscita V_{out}. Di contro, quando è applicato un livello logico 1 al gate, il transistore si comporta come un circuito elettrico aperto e il segnale analogico di ingresso non appare sulla linea di uscita V_{out}. Lo scopo del decodificatore è quello di controllare quale transistore sia in stato di bassa resistenza e in tal modo di controllare quale segnale analogico appare su V_{out}. Il decodificatore è progettato in modo tale che solo uno dei suoi segnali di uscita può essere al livello logico 0 in un qualsiasi momento. Perciò il circuito di *figura 4.3* funziona come una rete di interruttori a controllo digitale per otto segnali analogici in ingresso. Cioè, il circuito commuta i segnali analogici sulla linea V_{out} sotto il controllo dei segnali digitali del decodificatore.

Funzionamento del decodificatore

Il funzionamento del decodificatore di *figura 4.3* è relativamente semplice. Per prima cosa, tutte le tre linee di attivazione devono essere nel loro stato vero o attivo, in modo che una delle uscite da 0 a 7 vada al livello logico 0. Perciò se non è applicato uno 0 logico sia ad $\overline{\text{E1}}$ che ad $\overline{\text{E2}}$ nello stesso istante in cui è applicato un livello logico 1 ad E3, nessuna delle uscite del decodificatore andrà al livello logico 0, indipendentemente dai livelli logici applicati agli ingressi selezionati A, B o C.

Nel decodificatore multiplexer tutte e tre le linee di abilitazione devono essere nel loro stato attivo prima che una linea d'uscita vada al livello logico 0. Quale linea d'uscita debba andare al livello logico 0

Se tutti gli ingressi di attivazione sono al loro livello vero o attivo, allora il codice binario sulle linee A, B e C determina quale linea di uscita andrà al livello logico 0. Se A, B e C sono tutte al livello logico 0, l'uscita 0 (l'equivalente decimale del numero binario 000) andrà al livello 0, portando in conduzione il transistore più alto in *figura 4.3*, il quale permette alla tensione V_1 di apparire su V_{out}. Analogamente, se le linee A, B e C sono tutte al livello logico 1, l'uscita 7 (7 è l'equivalente decimale del codice binario 111) andrà al livello logico 0. Questo porterebbe il transistore più basso in stato di bassa resistenza e la tensione V_8 verrebbe collegata alla linea V_{out}. Perciò questo circuito permette ad un elemento digitale di controllo di mandare elettricamente sulla linea di uscita un qualsiasi segnale analogico di ingresso.

Un circuito di questo tipo potrebbe essere impiegato in un sistema che raccoglie auviene determinato dal codice binario inviato sulle linee A,B e C.

tomaticamente dei dati da otto o più ingressi diversi. Gli otto ingressi di tensione analogica vengono selezionati uno alla volta dal decodificatore sotto il controllo del sistema di controllo digitale. In un tale tipo di sistema la tensione analogica V_{out} potrebbe essere applicata ad un convertitore A/D per ottenere un segnale digitale da far elaborare ad un sistema digitale.

Ricerca ed eliminazione dei guasti

Anche se occasionalmente anche un circuito integrato MSI come il decodificatore si guasta, il guasto più probabile si verifica di solito in un collegamento col decodificatore. Perciò, quando c'e' un guasto, la prima cosa da fare è verificare con molta cura i circuiti che circondano il decodificatore e i loro collegamenti.

Esistono solo pochi tipi differenti di guasti che possono verificarsi nel circuito di *figura 4.3*; la maggior parte di questi hanno a che fare con problemi di collegamento, sebbene i guasti di un componente debbano comunque essere considerati. Se, per esempio, viene applicato un 1 logico a ciascuna delle linee da A_{11} a A_{15} e ad E3, e viene applicato uno 0 logico ad $\overline{E2}$, dovrebbe apparire un'uscita bassa sulla linea di uscita corrispondente al codice di selezione a tre bit su A_8, A_9 e A_{10}. Applicando tutti gli otto possibili codici di selezione a tre bit si può verificare completamente il funzionamento del decodificatore. Inoltre, ponendo una delle linee di attivazione al suo livello inattivo, tutte le uscite del decodificatore dovrebbero andare al livello logico 1. Se le uscite del decodificatore non hanno i livelli corretti, il guasto dovrebbe trovarsi nello stesso decodificatore o nei suoi collegamenti di ingresso, oppure nell'ingresso della stessa porta NAND o nei collegamenti di ingresso di quest'ultima. Questi guasti possono essere isolati utilizzando metodi precedentemente descritti; quindi il componente difettoso può essere sostituito o il problema nel cablaggio può essere riparato.
Se il decodificatore sta funzionando correttamente in risposta ai segnali di controllo in ingresso e non c'è V_{out} per nessun codice di selezione a 3 bit (supponendo che siano presenti le tensioni da V_1 a V_8), ci dovrebbe essere un circuito aperto tra il punto comune di collegamento dei transistori e il terminale di V_{out}. Se una sola tensione di ingresso non è presente su V_{out} quando viene utilizzato per quell'ingresso il codice di selezione a 3 bit, ma ciascuna delle altre tensioni di ingresso è presente su V_{out} quando ognuna di esse viene selezionata, il guasto è dovuto a: (1.) un transistore difettoso, o (2.) un filo aperto tra il transistore e la tensione di ingresso, o (3.) un filo aperto tra il transistore e il collegamento di V_{out}, o (4.) un filo aperto tra il gate del transistore e l'uscita del decodificatore. Osservando il tipo di difetto di funzionamento, il guasto può essere circoscritto ad una piccola parte del circuito in modo tale che si potranno effettuare prove solo su quella parte.

Multiplexer digitale

A seconda del livello logico del segnale di controllo, un multiplexer fa apparire sulle linee d'uscita i segnali binari a 8 bit provenienti o dal registro ad 8

Un'altra configurazione comune dei circuiti digitali MSI è il multiplexer digitale mostrato in *figura 4.4*. Lo scopo di questo multiplexer è quello di funzionare come un interruttore digitale per collegare le linee di uscita da A_0 ad A_7 o con gli 8 segnali binari provenienti dal registro ad 8 bit o con gli 8 segnali binari provenienti dal contatore ad 8 bit. Quale serie di otto segnali appaia sulle linee di uscita dipende dallo stato del segnale di controllo REF. Se REF è al livello logico 0, le uscite del registro vengono inviate alle linee di uscita. Se REF è al livello logico 1, le uscite del contatore verranno trasmesse alle linee da A_0 a A_7.

bit o dal contatore ad 8 bit.

Questa figura introduce due tipi di circuito MSI che non sono ancora stati trattati: il multiplexer ottale due-a-uno e il contatore a 8 bit. Il multiplexer è un circuito combinatorio e il contatore è un circuito sequenziale.

Come dice il nome stesso, la funzione del contatore è quella di contare. Esso conta il numero di impulsi che appaiono sulla linea INC. Si ricordi che un impulso è un segnale digitale che cambia (compie una transizione) da un livello logico ad un altro in un certo istante e compie la transizione opposta poco tempo dopo. Generalmente, un contatore si incrementa quando avviene una delle due transizioni. Nel circuito di *figura 4.4*, il conteggio aumenta ogni volta che il segnale INC compie una transizione da un livello basso a un livello alto. Il numero massimo che si può contare è 2^8 o 256 in quanto il contatore ha un codice di uscita a 8 bit. Perciò il contatore può contare da uno stato contenente 8 zeri ad uno stato contenente 8 uni per un totale di 256 differenti codici binari. Il contatore viene detto essere modulo 256 in quanto ad ogni 256esimo impulso il contatore si azzera e ricomincia a contare da capo.

Il registro a 8 bit in *figura 4.4* è un registro a memorizzazione sincrono. I segnali sulle linee da AD_0 ad AD_7 vengono immagazzinati nel registro quando il segnale di controllo ALE effettua una transizione da un livello basso ad un livello alto. I valori memorizzati quindi appaiono sull'uscita del registro e vengono applicati al multiplexer.

Ricerca ed eliminazione dei guasti

Due tipi di guasti che

È possibile avere diversi difetti di funzionamento in un circuito del tipo mostrato in *figura 4.4*. Uno di questi è che potrebbe non cambiare nessuna delle uscite compre-

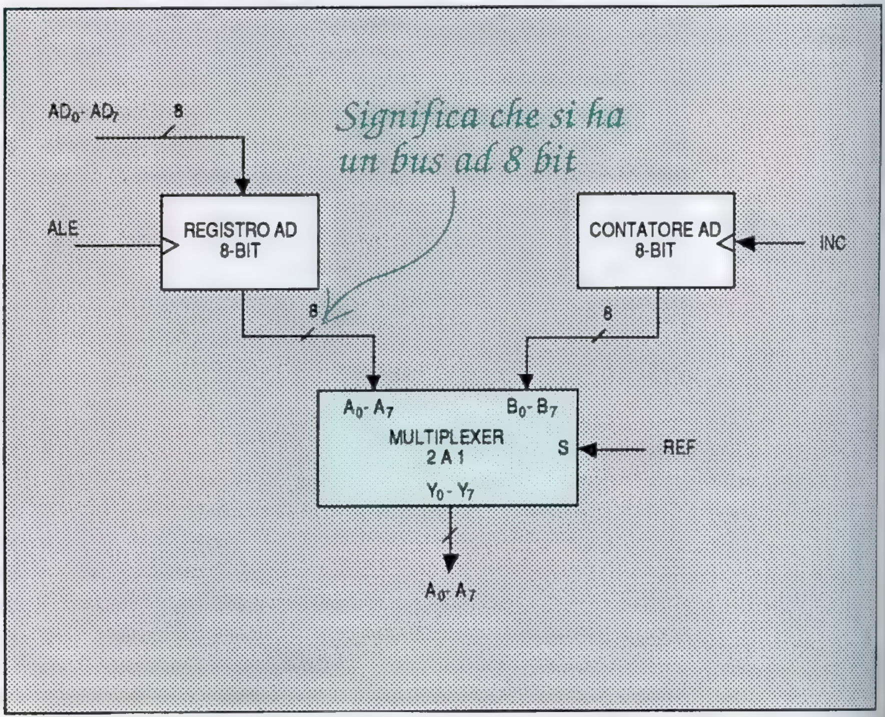


Figura 4.4. Esempio di circuito con multiplexing digitale

comunemente si verificano in un multiplexer sono: 1) nessuna delle uscite cambia, indipendentemente dallo stato del segnale di controllo; 2) un'uscita non cambia anche se gli ingressi corrispondenti sono a livelli logici differenti.

se tra A_0 ed A_7 indipendentemente dallo stato di REF. Questo implicherebbe un guasto nel multiplexer oppure un circuito aperto o un corto circuito sulla linea REF. Analogamente, se le uscite del registro ad 8 bit non cambiano mai, si ha che o il registro è guasto oppure c'è un circuito aperto o un corto circuito sulla linea ALE. Tutto ciò può essere applicato anche al contatore e alla sua linea di controllo INC.

Se il codice del contatore aumenta di 1 per ogni transizione da un livello basso a un livello alto sulla linea INC e se questo codice viene osservato sugli ingressi compresi tra B_0 e B_7, il contatore e i suoi fili di collegamento funzionano in modo corretto. Analogamente, se le uscite del registro cambiano per assumere gli stessi valori degli ingressi sulle linee da A_0 ad A_7 quando si verifica una transizione da un livello basso a un livello alto sulla linea ALE, e questi stessi valori vengono osservati sugli ingressi A_0 ed A_7 del multiplexer, il registro e il suo cablaggio stanno funzionando correttamente. Sotto queste condizioni, il guasto che non provoca alcun cambiamento sulle linee di uscita A_0 - A_7 potrebbe essere dovuto ad un multiplexer difettoso oppure a dei circuiti aperti nel cablaggio o nei collegamenti del multiplexer.

Un altro possibile difetto di funzionamento è quello per cui un'uscita del multiplexer non cambia sebbene i suoi corrispondenti ingressi siano a differenti livelli logici. Per esempio, se il registro manda in uscita 8 zeri e il contatore 8 uni, le uscite del multiplexer dovrebbero passare da tutti zeri a tutti uni in quanto REF è passato dal livello logico 0 al livello logico 1. Se, in tali condizioni di prova, tutte le uscite del multiplexer eccetto Y_1 passano da 0 ad 1, la causa potrebbe essere una delle seguenti: (1.) un difetto nel circuito di ingresso del multiplexer B_1, oppure (2.) un difetto nel circuito di uscita Y_1, oppure (3.) un corto circuito sul filo di uscita Y_1 oppure (4.) un circuito aperto sul filo di ingresso B_1. Dopo aver provato con misurazioni sul componente isolato o con tecniche di rilevazione del segnale che il multiplexer funziona in modo corretto, l'analisi per la ricerca del guasto può essere limitata ai collegamenti e al cablaggio sui terminali di B_1 e Y_1. Sapendo che il guasto è limitato ad una piccola area dell'intero circuito, il riparatore può ridurre considerevolmente il tempo delle prove rispetto a quello che gli servirebbe se ogni filo ed ogni componente fosse da esaminare nei dettagli. Naturalmente, questo discorso vale per ogni ricerca ed eliminazione di guasti.

GUASTI NEI CIRCUITI DIGITALI LSI (MEMORIE A SOLA LETTURA)

I tipi più complicati di circuiti combinatori sono la memoria a sola lettura (ROM) e la rete logica programmabile (PLA da Programmable Logic Array). Entrambi i tipi di circuito hanno migliaia di transistori all'interno di un singolo circuito integrato; perciò essi sono conosciuti come circuiti integrati su larga scala (LSI). Entrambi assolvono la funzione di memorizzare le informazioni tramite lo schema di cablaggio esistente tra le porte all'interno del dispositivo.

Le ROM sono circuiti combinatori complessi utilizzati per memorizzare

Il semplice esempio di ROM mostrato in *figura 4.5* ha solo 4 posizioni di immagazzinamento (4 porte NAND di ingresso indicate come G1-G4) e 4 bit per posizione (4 porte NAND di uscita indicate come G5-G8). L'interconnessione tra le uscite di G1-G4 e gli ingressi di G5-G8 determina quale codice binario viene immagazzinato in ogni posizione di memoria. Le porte G1-G4 costituiscono un decodificatore di linee curve di funzioni, tabelle di conversioni di codici o altre tabelle di costanti utilizzate in uno specifico sistema. Le ROM tipiche forniscono una parola di 8 bit (un byte) quando vengono indirizzate con il codice corretto.

m - 2^m. Nell'esempio di *figura 4.5* m è 2 e quindi G1-G4 costituiscono un decodificatore 2 a 4. Solo due linee di indirizzo, A_0 ed A_1 sono necessarie per indirizzare le quattro posizioni. Quando entrambe queste due linee sono al livello logico 0, solamente G1 ha degli 1 su entrambi i terminali di ingresso cosicché solamente G1 produrrà un segnale al livello logico 0. Questo 0 logico provocherà un'uscita 1 sulle porte NAND con le quali è collegato, che, in questo caso, sono G7 e G8. Perciò il codice di uscita a 4 bit è 0011 in risposta ad un codice di ingresso di 00. Analogamente, quando il codice di indirizzo è 01 (0 su A_1 e 1 su A_0), solo G2 ha degli 1 su entrambi i suoi ingressi e solo quella porta emetterà in uscita uno 0 logico. Questo 0 logico porterà G5 e G6 ad avere come uscita 1. Perciò, verrà generato un codice di uscita di 1100 ogniqualvolta viene applicato un codice di indirizzo di 01 agli ingressi della ROM. Il lettore può verificare che un codice di indirizzo di 11 provocherà un codice di uscita di 0110 mentre un codice di indirizzo di 10 produrrà un codice di uscita di 1101. Poiché il fatto che venga generato un certo codice di uscita in risposta ad un codice di indirizzo è fissato al momento della fabbricazione, lo schema d'immagazzinamento non può essere cambiato; cioè, non è possibile scrivere in memoria. Ecco perché questa memoria viene chiamata memoria a sola lettura.

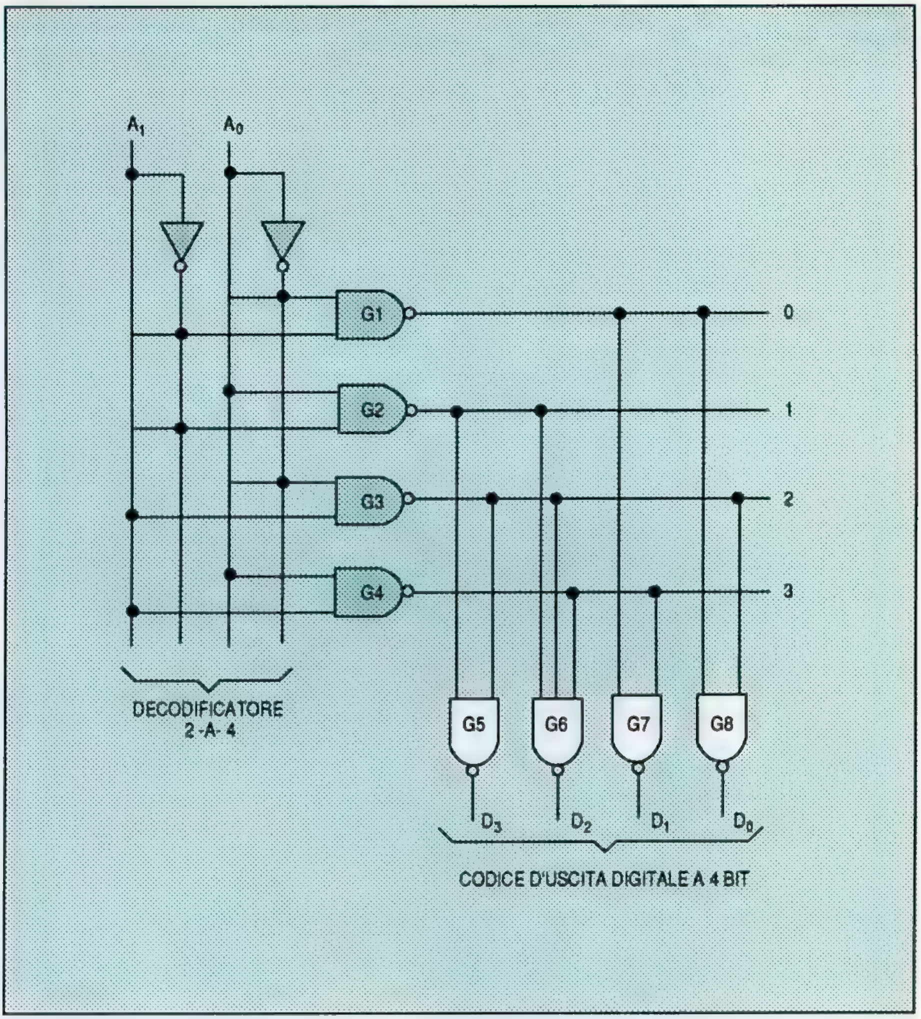
Il semplice esempio in *figura 4.5* con soli 4 bit serve solamente come spiegazione. Normalmente, una ROM fornisce migliaia di locazioni a 8 bit. Un esempio di una piccola ROM a 256 byte che può servire come dispositivo di memoria ad alta velocità è mostrato in *figura 4.6a*. Per i microcomputer, un byte consiste normalmente di 8 bit, perciò questo dispositivo fornisce 256 locazioni con 8 bit di memoria ciascuna. Poiché 256 = 2^8, sono richiesti 8 ingressi dalle linee di indirizzo, in modo che un sistema digitale esterno, come un microprocessore, possa selezionare una qualsiasi delle 256 locazioni per l'uscita. Per eliminare la confusione che troppe linee avrebbero provocato nel disegno, viene mostrato per ogni uscita delle porte NAND un solo ingresso, ma i puntini corrispondenti alle interconnessioni dei fili indicano quale porta ha un 1 in uscita per ogni linea attiva (0). I codici immagazzinati in memoria sono mostrati in *figura 4.6b* rispetto agli ingressi di indirizzo A_0 - A_8

L'applicazione di tali memorie ROM è quella di memorizzare curve di funzioni, tabelle di conversione di codici o altre tabelle di costanti utilizzate in un particolare sistema. Generalmente, se si deve utilizzare la ROM in sistemi di microcalcolatori, viene collegato un buffer ottale a tre stati tra l'uscita delle porte NAND e i piedini del circuito integrato, come mostrato. Il buffer viene abilitato da un segnale di controllo indicato in figura come $\overline{CS}$ (Chip select not).

Un suono può essere generato memorizzandone l'inviluppo in un codice digitale all'interno della ROM e indirizzando la locazione della ROM nella corretta sequenza e alla corretta frequenza.

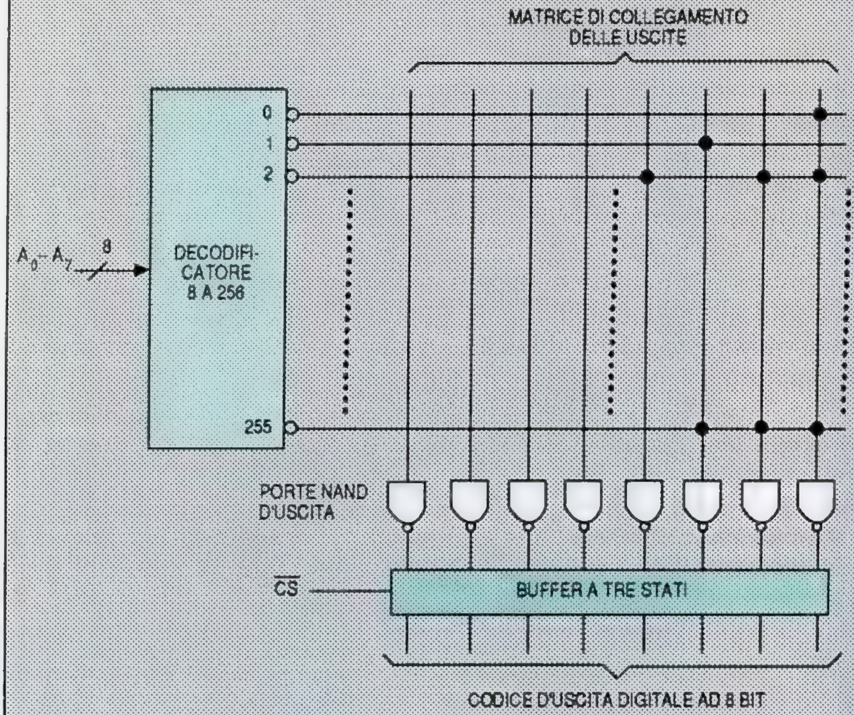
In *figura 4.7* è mostrato un esempio di applicazione di una memoria ROM come generatore di suoni. L'indirizzo è fornito da un contatore a 8 bit il cui conteggio aumenta di 1 ogniqualvolta si verifica una transizione da un livello basso ad un livello alto sulla linea INC. Perciò, se il segnale INC compie una transizione di questo tipo ogni millisecondo, il valore del conteggio aumenterà di 1 ogni millisecondo, il che dà 1000 incrementi per secondo. Questo codice a 8 bit viene inviato alle linee di indirizzo A_0 - A_7 della ROM in modo tale che la ROM produca il codice memorizzato che corrisponde a ciascun indirizzo. Perciò il codice di uscita su D_0 - D_7 cambia ogni millisecondo. Il convertitore digitale - analogico (D/A) converte il codice a 8 bit proveniente dalla ROM in uno dei 256 differenti livelli di tensione analogica. (Si ricordi che il convertitore D/A è stato trattato nel capitolo 1.) Questo segnale analogico viene am-

Figura 4.5. Semplice ROM a 4 locazioni con 4 bit per locazione



plificato da un amplificatore audio e viene applicato ad un altoparlante. Memorizzando nella ROM l'inviluppo del suono che deve essere generato e fornendo impulsi al contatore con una frequenza appropriata, si può far emettere all'altoparlante il suono desiderato. Poiché il contatore ripeterà continuamente il ciclo attraverso i codici compresi tra 0000 0000 e 1111 1111, lo stesso suono verrà ripetuto in continuazione. Se nella ROM è codificata una funzione seno (si ricordi l'onda sinusoidale del capitolo 1), verrà prodotto un tono quasi puro. Fornendo lo schema per la pronuncia di una lettera o di una parola e utilizzando la frequenza adatta (piuttosto che una frequenza costante) sul segnale INC, il circuito potrebbe sintetizzare suoni simili alla voce umana. Quest'ultima applicazione, comunque, richiederebbe una capacità della ROM maggiore di quella mostrata in *figura 4.7*.

Figura 4.6. Struttura di una ROM a 256 byte


MATRICE DI COLLEGAMENTO DELLE USCITE
0
1
2
255
A_0 - A_7
8
DECODIFICATORE 8 A 256
PORTE NAND D'USCITA
$\overline{CS}$
BUFFER A TRE STATI
CODICE D'USCITA DIGITALE AD 8 BIT

a. Circuito

INDIRIZZO A_0 - A_7	CODICE D'USCITA
00000000	00000001
00000001	00000100
00000010	00001011
.	.
.	.
.	.
11111111	00000111

b. Codice Memorizzato

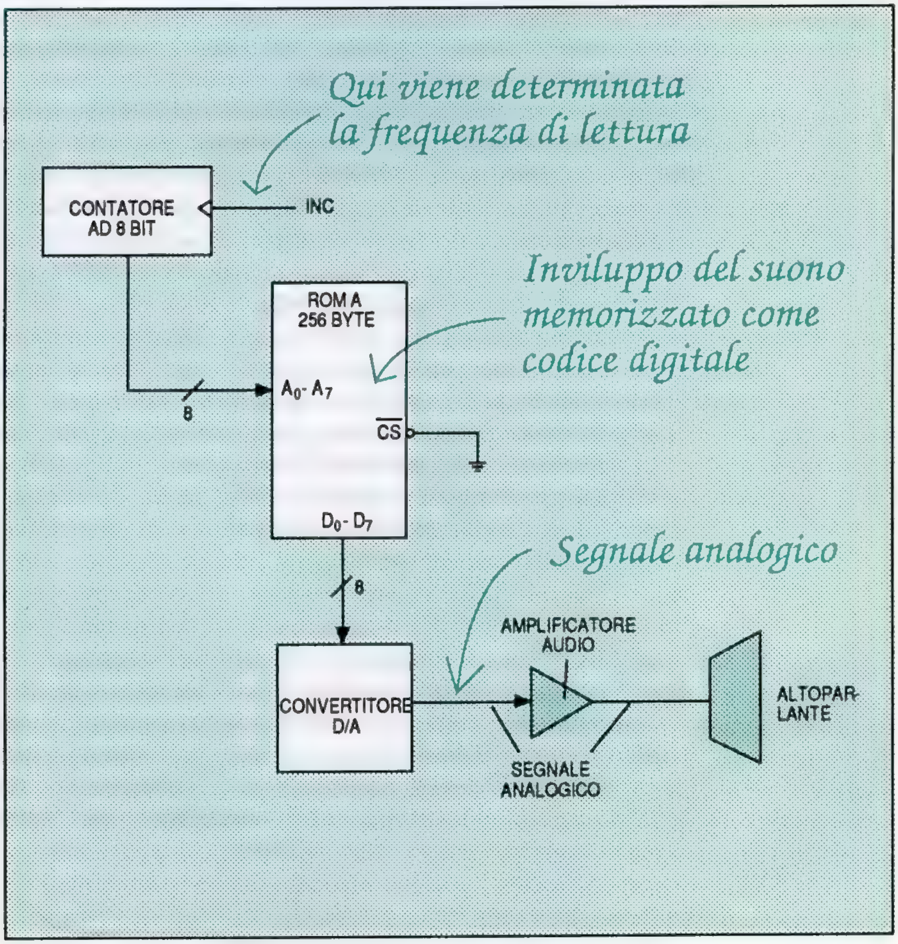
Ricerca ed eliminazione dei guasti

Il circuito di *figura 4.7* potrebbe guastarsi in diversi modi. Si potrebbero osservare malfunzionamenti quali un suono sbagliato, un suono troppo debole o nessun suono proveniente dall'altoparlante. Poiché si tratta di un sistema sia analogico che digitale (sistema misto), devono essere utilizzati metodi di ricerca ed eliminazione dei guasti sia analogici che digitali.

Un sistema ibrido, come l'esempio del generatore di suoni, richiede tecniche di ricerca dei guasti sia analogiche che digitali. Dapprima viene utilizzato il metodo di

Un approccio potrebbe essere quello di cominciare con un procedimento di rilevazione del segnale per verificare il funzionamento della parte analogica del circuito. Ciò può essere fatto isolando il piedino di uscita del convertitore D/A e iniettando un'onda sinusoidale di 1000 hertz (o di ogni altra frequenza audio adatta) nell'altoparlante attraverso un amplificatore opportuno. Si dovrebbe sentire un tono chiaro e corretto se l'altoparlante e i suoi fili di collegamento non sono guasti. Se l'altoparlante funziona, il segnale audio dovrebbe essere iniettato nell'ingresso dell'amplificatore audio. Se, anche in questo caso, si ascolta un tono chiaro, l'amplificatore funziona correttamente. Se, attraverso il procedimento di rilevazione dèl segnale, si sco-

Figura 4.7. Semplice applicazione della ROM in un generatore di suoni


Qui viene determinata la frequenza di lettura
CONTATORE AD 8 BIT
INC
ROM A 256 BYTE
Inviluppo del suono memorizzato come codice digitale
A0 - A7
8
CS
D0 - D7
8
Segnale analogico
AMPLIFICATORE AUDIO
CONVERTITORE D/A
ALTOPARLANTE
SEGNALE ANALOGICO

rilevazione del segnale nel circuito analogico per verificare il suo funzionamento. Se funziona correttamente, restano da provare il contatore e la ROM.

pre che l'altoparlante o l'amplificatore audio o il loro cablaggio di interconnessione sono difettosi, il guasto può essere riparato o il componente guasto sostituito. Se la parte analogica del circuito funziona, allora devono essere effettuate delle misurazioni sulla parte digitale.

Il componente più semplice da controllare nella parte digitale è il contatore. Fornendo impulsi sulla linea INC attraverso un impulsatore logico, i livelli di tensione sulle uscite del contatore dovrebbero cambiare in modo ordinato. Una sonda logica sulla linea A0 dovrebbe lampeggiare con una frequenza che è metà di quella della linea INC. La linea A1 dovrebbe indicare una frequenza che è metà di quella osservata sulla linea A0 e così via fino alla linea A_7. Utilizzando una clip logica e un impulsatore sulla linea INC, i bit che rappresentano l'uscita del contatore dovrebbero contare da 0000 0000 a 1111 1111 con incrementi di 1. Se le spie sulla clip logica hanno luci fisse quando è applicato il generatore di impulsi alla linea INC, la linea INC è aperta o ha subito un corto a massa oppure il circuito del contatore è guasto. Se il contatore sembra funzionare correttamente ad eccezione del fatto che una linea di uscita è sempre ad un livello basso o alto, il circuito del contatore è guasto oppure c'è un'apertura o un corto circuito sulla quella linea di uscita. Una volta verificato il buon funzionamento del contatore e del suo cablaggio, si potrà considerare il funzionamento della ROM.

Se si conosce lo schema di memorizzazione della ROM, un semplice approccio consiste nell'applicare l'impulsatore alla linea INC e una clip logica sul circuito della ROM. Gli ingressi della ROM dovrebbero mostrare lo schema binario del contatore che cambia con la stessa frequenza degli impulsi sulla linea INC. Le luci sulle linee di uscita da D_0 a D_7 dovrebbero cambiare secondo uno schema che è rappresentativo della tabella della funzione memorizzata nella ROM. Per esempio, se nella ROM è memorizzata un'onda triangolare o sinusoidale, in uscita si dovrebbe avere uno schema binario periodico che inizia da valori di 0 o vicini allo 0, gradualmente cresce fino ad avere tutti 1, quindi decresce gradualmente fino ad avere nuovamente valori vicini allo 0. Se tutte le uscite sono bloccate a 1 o a 0 anche quando i codici sulle linee da A_0 ad A_7 cambiano continuamente, i guasti più probabili sono quelli che riguardano i collegamenti di $\overline{CS}$, della massa o di V_{cc} (alimentatore). Anche una ROM guasta produrrebbe però effetti simili. I controlli per vedere se esistono circuiti aperti o corto circuiti nel cablaggio possono essere effettuati con le solite tecniche di verifica della continuità dei fili o di misurazione della tensione. Se non ci sono guasti nel cablaggio, la ROM avrebbe bisogno di essere provata in un microcomputer o utilizzando un circuito di test che permetterà di confrontare la configurazione di uscita della ROM con la configurazione di codificazione nota per quel tipo di memoria.

È molto noioso verificare il funzionamento di ciascuna locazione della ROM manualmente. Un metodo più semplice utilizza un microcomputer per indirizzare

Una tecnica per verificare il codice e il funzionamento di una ROM è quella di controllare ogni locazione di memoria in essa con un apposito circuito di prova. Con questo tipo di circuito, l'esatta configurazione di 0 e 1 memorizzati in ciascuna locazione può essere verificata effettuando misurazioni di tensione piedino per piedino o utilizzando una clip logica, ma questa è un'operazione lunga. Un procedimento più conveniente è quello di usare un microcomputer che provi il funzionamento della ROM leggendo ciascuna locazione di memoria bit per bit e confrontando il codice con quello (noto) che si suppone sia in essa. Questo metodo è particolarmente utile per ROM che hanno migliaia di locazioni di memoria.

ciascuna locazione della ROM e confrontare l'uscita con un codice noto.

Se tutte le uscite della ROM tranne una sembrano lavorare correttamente, i guasti possibili sono un difetto di cablaggio su quel piedino o un circuito di uscita della ROM guasto su quella linea.

Se la combinazione della ROM e del contatore funziona correttamente e produce l'esatta sequenza di codici sulle linee da D_0 a D_7 alla frequenza determinata da INC, l'unica possibilità che resta è quella di un guasto nel convertitore D/A o nel cablaggio ad esso associato.

Il convertitore D/A converte il codice di ingresso a 8 bit in un'uscita di livello analogico corrispondente. Se il cablaggio all'interno e all'esterno del convertitore D/A è risultato funzionante ad una verifica, il guasto dovrebbe essere probabilmente nel convertitore D/A oppure nei suoi collegamenti con l'alimentatore o la massa.

Il convertitore D/A viene posto sotto test introducendo codici digitali provenienti dal sistema e misurando l'uscita con un voltmetro.

Un semplice modo per provare il convertitore D/A consiste nell'utilizzare il circuito di *figura 4.7* per mandare codici digitali nel convertitore D/A mentre si misura l'uscita risultante con un voltmetro. Ad ogni modo, per osservare un qualsiasi cambiamento nella tensione con un voltmetro occorre che la frequenza degli impulsi INC (frequenza del verificarsi di transizioni da un livello basso a un livello alto) sia ridotta a un valore basso. Per esempio, un impulso INC da 2 a 10 impulsi per secondo provocherebbe un cambiamento dell'ingresso e dell'uscita del D/A abbassandone la frequenza in modo tale da poterla osservare con un voltmetro.

Controllando il codice digitale di ingresso con una clip logica e la tensione analogica di uscita con un voltmetro, sarebbe possibile verificare il funzionamento del convertitore. Per esempio, quando tutte le spie sulla clip sono spente (tutti gli ingressi sono 0), il voltmetro dovrebbe indicare il valore minimo. Quando gli ingressi più significativi (i bit A_7 e A_6) sono al livello 1, il voltmetro dovrebbe indicare un valore alto. L'indicazione sul voltmetro dovrebbe variare tra questi livelli bassi ed alti all'aumentare del codice binario da 0000 0000 a 1111 1111 nel modo che è indicato dal cambiamento della configurazione di luci accese e spente sulla clip logica. Se la tensione rimane sui valori più alti, allora una delle linee digitali di ingresso tra la ROM e il convertitore per i bit più significativi di ingresso (A_7 e A_6) deve essere fissa al livello 1, oppure lo stesso convertitore è guasto. Se la tensione resta invece su valori bassi, queste stesse linee di ingresso devono essere fisse a 0, oppure lo stesso convertitore è guasto. In entrambi i casi, sostituire il convertitore o correggere i guasti nel cablaggio dovrebbe bastare a riparare il sistema.

ESEMPIO DI CIRCUITO CODIFICATORE MSI

La funzione del codificatore è totalmente opposta a quella del decodificatore. Esso converte un singolo ingresso di linea in un codice d'uscita a tre bit che è unico per la linea d'ingresso.

Un circuito MSI che non è ancora stato trattato è il codificatore. Per introdurre questo dispositivo e discutere un ulteriore esempio di ricerca ed eliminazione dei guasti in un sistema MSI verrà considerato l'esempio di *figura 4.8*. Molto probabilmente questo tipo di circuito si troverà in un microcomputer. Fondamentalmente il codificatore compie la funzione inversa del decodificatore. Si ricordi che un decodificatore converte un codice ad m bit in 2^m linee di segnale. Nell'esempio di *figura 4.8* è usato ancora il decodificatore 3 a 8. In questo esempio, una delle linee di uscita del decodificatore, da G_0 a G_7, va al livello logico 0 quando viene selezionata dal codice a 3 bit sulle linee di ingresso A, B e C e quando l'ingresso di abilitazione $\overline{E}$ è anch'esso allo 0 logico.

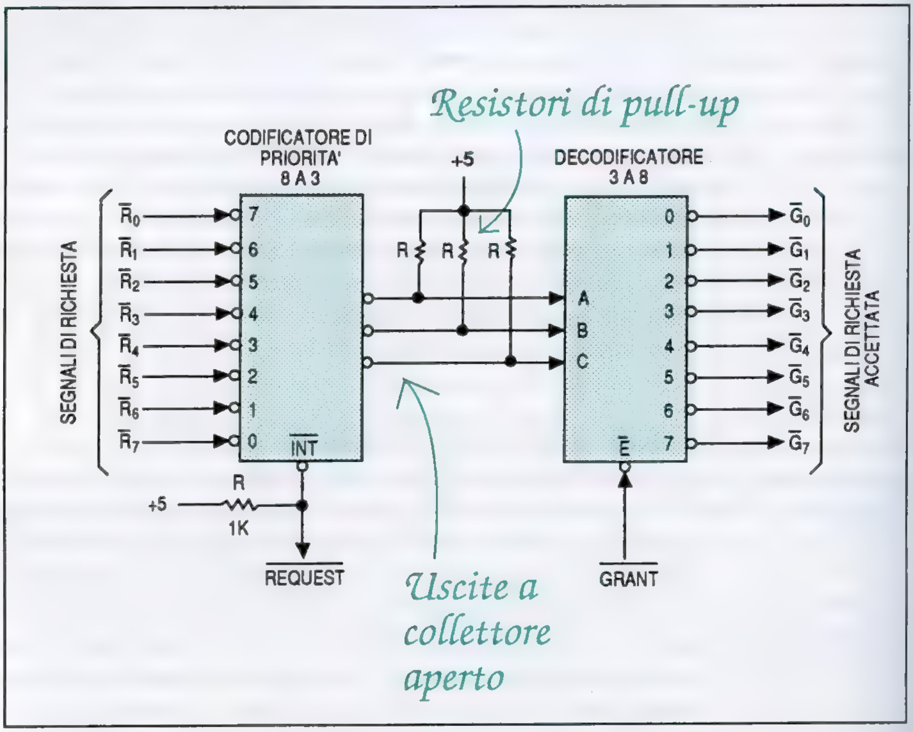
Il codificatore genera un codice binario ad m bit che corrisponde alla linea di ingresso al livello logico 0. Il codificatore utilizzato nel caso specifico di *figura 4.8* è un codificatore a priorità che determina quale di due o più ingressi al livello logico 0 nello stesso istante produrrà un'uscita. Esso dovrà fornire un codice a 3 bit che corrisponde a una sola di queste linee. Lo schema di priorità è prestabilito e non può essere cambiato. In un'applicazione, il modo in cui i circuiti esterni sono collegati con il codificatore determinerà quale dei circuiti esterni riceverà la priorità più alta. Nell'esempio di *figura 4.8*, gli ingressi sono numerati dalla priorità più bassa (ingresso 0) alla priorità più alta (ingresso 7). Perciò, se c'è uno 0 logico simultaneamente su entrambe le linee 0 e 5, il codice a 3 bit fornito dal codificatore sarà quello che corrisponde alla linea attiva 5.

Alcuni circuiti logici integrati hanno un'uscita con transistore a collettore aperto che richiede un resistore collegato tra l'uscita e l'alimentazione.

Queste uscite sono diverse per una particolarità che non è ancora stata trattata. Esse dispongono di una cosiddetta uscita a collettore aperto, il che richiede che sia collegato un resistore tra ogni uscita e l'alimentazione di 5 volt. La maggior parte dei dispositivi digitali come il decodificatore hanno questo resistore al loro interno, perciò le uscite del decodificatore non richiedono resistori esterni.

Le uscite del codificatore sono invertite e ciò viene indicato dai cerchietti su ogni uscita. Perciò, se l'ingresso 7 è allo 0 logico, il codice interno a 3 bit che corrisponde al numero decimale 7 è 111. Comunque, l'inversione o il complemento di 111 è 000; perciò viene inviata alle tre linee di uscita la configurazione di bit 000. Così, quando alla linea $\overline{R}_0$ (ingresso 7) viene applicato uno 0 logico, l'uscita del codifica-

Figura 4.8. Esempio di applicazione di un codificatore-decodificatore


Resistori di pull-up
CODIFICATORE DI PRIORITA' 8 A 3
+5
DECODIFICATORE 3 A 8
SEGNALI DI RICHIESTA
$\overline{R}_0$
$\overline{R}_1$
$\overline{R}_2$
$\overline{R}_3$
$\overline{R}_4$
$\overline{R}_5$
$\overline{R}_6$
$\overline{R}_7$
7
6
5
4
3
2
1
0
$\overline{INT}$
R
R
R
A
B
C
0
1
2
3
4
5
6
7
$\overline{E}$
$\overline{G}_0$
$\overline{G}_1$
$\overline{G}_2$
$\overline{G}_3$
$\overline{G}_4$
$\overline{G}_5$
$\overline{G}_6$
$\overline{G}_7$
SEGNALI DI RICHIESTA ACCETTATA
R
+5
1K
$\overline{REQUEST}$
Uscite a collettore aperto
$\overline{GRANT}$

tore sarà 000 e l'uscita $\overline{\text{INT}}$- sarà uno 0 logico. Ciò si verifica indipendentemente dai segnali binari applicati agli altri ingressi in quanto la linea $\overline{R}_0$ ha la priorità più alta. Gli ingressi e le uscite sono riassunte nella *tabella 4.1*. Quando una linea di ingresso riceve uno 0 logico, l'uscita $\overline{\text{INT}}$ passa allo 0 logico. Solo se tutte le linee di ingresso sono al livello logico 1 l'uscita INT sarà al livello logico 1.

Ora che il funzionamento di ogni singolo componente del circuito di *figura 4.8* è stato compreso può essere determinato il funzionamento generale del circuito. Il caso più semplice si presenta quando solo uno dei segnali di richiesta è attivo, cioè allo 0 logico. Si supponga, per esempio, che solo il segnale di richiesta $\overline{R}_3$ sia uno 0 logico e che tutti gli altri segnali di richiesta siano degli 1 logici. Il codificatore a priorità produrrà uscite di 0 su INT e di 011 sulle linee del codice di uscita a 3 bit (vedi *tabella 4.1*).

Il sistema di cui questo circuito è una parte controlla l'uscita $\overline{\text{INT}}$ per mezzo della linea $\overline{\text{REQUEST}}$. Quando esso individua il livello logico 0, sa che un ingresso di richiesta è attivo. Al momento giusto, esso risponde mettendo un valore basso sulla linea $\overline{\text{GRANT}}$ per abilitare il decodificatore. A quel punto la linea di uscita $\overline{G}_3$ del decodificatore andrà allo 0 logico. Questo 0 verrà usato dal sistema per attivare il sottosistema che aveva abbassato il valore sulla linea di richiesta $\overline{R}_3$ per avere un servizio. Comunque, se un altro sottosistema avesse richiesto funzioni di servizio contemporaneamente, cosicché due linee di richiesta sarebbero state attive a un livello basso contemporaneamente, il codificatore a priorità avrebbe prodotto un valore basso su $\overline{\text{INT}}$ e soltanto il codice a 3 bit che corrisponde a quella delle due richieste che ha priorità più alta.

Tabella 4.1. Ingressi ed uscite del codificatore

Segnale di Richiesta Attivo	Corrispondente Ingresso del Codificatore Attivo	Codice d'Uscita agli Ingressi A B C del Decodificatore
$\overline{R}_0$	7	0 0 0
$\overline{R}_1$	6	0 0 1
$\overline{R}_2$	5	0 1 0
$\overline{R}_3$	4	0 1 1
$\overline{R}_4$	3	1 0 0
$\overline{R}_5$	2	1 0 1
$\overline{R}_6$	1	1 1 0
$\overline{R}_7$	0	1 1 1

Ricerca ed eliminazione dei guasti

Così come per il decodificatore, la ricerca dei guasti nel circuito del codificatore dovrebbe cominciare con un esame delle interconnessioni col codificatore.

Una volta compreso il funzionamento generale del circuito di figura 4.8, è possibile considerare alcuni sintomi di malfunzionamento e determinare la loro probabile causa. Ecco il primo problema da considerare: durante il funzionamento del sistema, le linee di riconoscimento $\overline{G}_2$, $\overline{G}_3$, $\overline{G}_6$ e $\overline{G}_7$ non vanno mai allo 0 logico anche se le linee di richiesta $\overline{R}_2$, $\overline{R}_3$, $\overline{R}_6$ o $\overline{R}_7$ sono la linea a priorità più alta che riceve lo 0 logico. Le altre linee di riconoscimento lavorano normalmente in risposta a 0 logici sulle loro proprie linee di richiesta. In che parte dell'intero circuito è situato con maggiore probabilità il guasto?

È poco probabile che il decodificatore e il codificatore si siano guastati in questo modo (anche se la possibilità esiste). Inoltre, non è ragionevole pensare che i fili che collegano gli ingressi R guasti o le uscite G guaste abbiano subito un guasto contemporaneamente e in modo tale da provocare il comportamento osservato del circuito. Perciò l'area in cui più probabilmente si è verificato il guasto è un circuito di ingresso o un circuito di uscita oppure un filo di interconnessione che è in qualche modo comune alla parte di circuito sotto analisi. Se si esaminano i codici a 3 bit che corrispondono alle condizioni di uscita errate e i codici delle uscite che invece sono funzionanti, come mostrati in *tabella 4.2*, si osserva una caratteristica comune.

Il bit che separa tutti i codici funzionanti da quelli non funzionanti è il bit di mezzo sulla linea B. Solo le uscite che hanno uno 0 sulla linea B lavorano correttamente. Apparentemente non è possibile portare la linea B al livello 1 in modo tale da generare le uscite che non funzionano. Perciò la linea B deve essere fissa a 0. Un modo per confermare ciò è richiedere $\overline{R}_2$. Se la linea B è fissa a 0, l'uscita $\overline{G}_0$ dovrebbe essere 0 in quanto l'uscita del codificatore dovrebbe essere 000 (vedi la *tabella 4.1*). Analogamente, richiedendo $\overline{R}_3$ risulterebbe un'uscita $\overline{G}_1$.

Se un controllo per la ricerca di corto circuiti o di circuiti aperti svolto attraverso le rilevazioni dei valori di tensione sui piedini del circuito integrato non rivela il problema, per determinare se il circuito integrato ha un guasto dovranno essere utilizzati i metodi di isolamento dei piedini, di verifica tramite una sonda logica e di rilevazione del segnale.

La causa più probabile per cui la linea B può essere bloccata a 0 potrebbe essere un'apertura nel circuito tra il terminale dell'alimentatore e l'ingresso B del decodificatore o l'uscita del codificatore. In quest'area sono compresi il resistore, il cablaggio e i punti di connessione. Un'altra causa probabile potrebbe essere un corto a massa lungo la linea B tra il decodificatore e il codificatore. Misurando il resistore ed effettuando controlli di continuità tra il terminale dell'alimentatore e la linea B si dovrebbe poter determinare la posizione di un'eventuale apertura verso l'alimentatore. Verificando la continuità lungo il filo dal codificatore al decodificatore si dovrebbe localizzare un'apertura in quel collegamento. Alimentare il circuito e misurare la tensione sulla linea B con uno 0 logico applicato all'ingresso $\overline{R}_0$ dovrebbe determinare la presenza di un corto a massa sul filo B. Se il cablaggio e il resistore funzionano correttamente, allora l'ingresso B del decodificatore o l'uscita B del codificatore devono avere un guasto interno. L'isolamento del piedino e le tecniche di misura con la sonda logica dovrebbero rivelare quale di questi componenti deve essere sostituito. Come in tutti i circuiti, ci sono molti punti dove è possibile che si verifichi un guasto e in questo esempio se ne è considerato uno soltanto. Il procedimento per isolare un guasto e ripararlo rimane comunque lo stesso. Una volta capito il principio di funzionamento del circuito e osservati i sintomi del guasto, il ragionamento basato sul funzionamento dovrebbe restringere la posizione del guasto ad una piccola parte del circuito in modo tale da ridurre le prove per la verifica del funzionamento ad

Tabella 4.2. Uscite con codice a tre bit

Uscite Corrette	Uscite Errate
0 0 0	0 1 0
0 0 1	0 1 1
1 0 0	1 1 0
1 0 1	1 1 1

un numero ragionevole. Se il riparatore dovesse effettuare tutte le possibili misurazioni in un circuito anche dei più semplici dei tipi trattati in questo capitolo, la quantità di lavoro e di dati da interpretare sarebbe enorme.

CHE COSA ABBIAMO IMPARATO?

1. I componenti combinatori fondamentali, che comprendono i buffer a tre stati, i decodificatori, i codificatori e le memorie ROM, possono essere combinati l'uno con l'altro e con porte semplici per svolgere funzioni di comune utilizzazione.
2. Abbiamo imparato come alcuni di questi circuiti funzionano e che è possibile individuare la maggior parte dei guasti più comuni che si verificano in essi attraverso il ragionamento e basandosi su semplici prove.
3. Il contatore binario è un circuito sequenziale. Questi circuiti sono più difficili da riparare perché il loro funzionamento dipende da una sequenza di segnali che spesso cambiano molto rapidamente e dalle relazioni temporali tra tali segnali.

Quiz del Capitolo 4

1. Un buffer a tre stati è un dispositivo che:
    a. Ha tre differenti linee di uscita.
    b. Ha tre differenti livelli logici.
    c. Ha uno stato di alta impedenza oltre ai normali livelli 0 e 1.
    d. È un tipo speciale di registro di memoria.

2. Perché un buffer a tre stati produca livelli 0 o 1 in uscita deve essere vero quanto segue:
    a. L'informazione deve essere stata memorizzata nel buffer.
    b. L'abilitazione di uscita deve essere vera.
    c. L'abilitazione di uscita deve essere falsa.
    d. Il segnale $\overline{OE}$ deve essere al livello logico 1.

3. Un decodificatore è un circuito MSI le cui linee di uscita sono controllate da:
    a. Un codice di selezione di ingresso binario.
    b. Una serie simile di linee di ingresso.
    c. Un'abilitazione di uscita.
    d. Un ingresso di clock.

4. Se il decodificatore mostrato in *figura 4.3* è abilitato con un codice di ingresso a 3 bit di 010, si avrà la seguente uscita:
    a. Solo la linea 5 sarebbe al livello logico 0.
    b. Solo la linea 5 sarebbe al livello logico 1.
    c. Solo la linea 2 sarebbe al livello logico 0.
    d. Solo la linea 2 sarebbe al livello logico 1.

5. Se un circuito multiplexer ha 4 linee di ingresso e una sola linea di uscita, a quanti bit deve essere il codice di selezione?
    a. 0
    b. 1
    c. 2
    d. 4

6. Un codificatore è un circuito MSI che:
    a. Provvede a trasmettere solo uno di due o più ingressi ad un'uscita.
    b. Seleziona una data linea di uscita sulla base di un codice binario di ingresso.
    c. Fornisce un codice di uscita che corrisponde alla linea vera di una serie di linee di ingresso.
    d. Permette di memorizzare un certo numero di bit binari.

7. Una ROM contenente 512 byte di memoria quante porte di uscita avrà?
    a. Nessuna
    b. 4
    c. 8
    d. 16
    e. 512

8. Se nell'esempio di *figura 4.8* solo gli ingressi compresi fra $\overline{R}_0$ e $\overline{R}_3$ producono uscite corrette, qual è il guasto più probabile nel circuito?
    a. La linea A bloccata a 0
    b. La linea A bloccata a 1
    c. La linea B bloccata a 0
    d. La linea C bloccata a 0
    e. La linea C bloccata a 1

9. Se il circuito di *figura 4.1* non risponde alla chiusura di nessuno degli interruttori S6, S7 ed S8, qual è il guasto più probabile?
    a. La linea tra C e gli interruttori ha subito un corto a massa.
    b. Sulla linea tra il punto C e il punto H c'è un'apertura.
    c. Tutti e tre gli interruttori sono guasti.
    d. La porta NAND è guasta.

10. Se in *figura 4.7* le uscite della ROM sono tutte indipendenti dagli ingressi della ROM e non producono una tensione al livello logico 1 o al livello logico 0, qual è il guasto più probabile?
    a. Il contatore è guasto.
    b. Il convertitore D/A è guasto.
    c. C'e' un'apertura nel collegamento $\overline{CS}$.
    d. I circuiti di ingresso alla ROM sono guasti.

PROBLEMI DI LOGICA SEQUENZIALE

SCOPO DEL CAPITOLO

Gli esempi dei capitoli precedenti si limitavano a trattare i guasti e i loro effetti nei circuiti combinatori. Questi circuiti hanno la caratteristica di avere la loro uscita direttamente in relazione con la combinazione di segnali di ingresso che è presente in ogni dato istante. Il tipo di logica che sarà considerato in questo capitolo e nei seguenti è la logica sequenziale. La maggior parte dei sistemi digitali di qualsiasi complessità, inclusi i sistemi dei microcomputer, utilizzano la logica sequenziale.

Si ricordi che lo stato delle uscite dei sistemi logici sequenziali è determinato sia da quello che è successo nel passato che dall'ingresso attuale. I componenti di memoria, come i flip-flop e i registri, formano dei sistemi sequenziali.

I sistemi a logica sequenziale sono caratterizzati dal fatto che le uscite e la sequenza di operazioni del sistema dipendono sia dagli ingressi presenti che dagli ingressi che erano presenti negli istanti immediatamente precedenti. In altre parole, ciò che accade all'uscita del sistema e il passo successivo da compiere nel funzionamento del sistema vengono determinati da ciò che si trova all'ingresso in quel momento e da ciò che vi si trovava nell'istante precedente. I sistemi a logica sequenziale sono composti da dispositivi di memoria quali i flip-flop, i registri o i circuiti di memoria a semiconduttore per memorizzare quello che è successo precedentemente. Sebbene i sistemi a logica sequenziale siano spesso molto più complicati di quelli a logica combinatoria per quanto riguarda la loro progettazione e il loro funzionamento, i procedimenti fondamentali utilizzati per la ricerca dei guasti nei sistemi a logica combinatoria si applicano altrettanto bene ai sistemi a logica sequenziale. Si ricordi che uno dei metodi di ricerca di guasti è quello di ragionare basandosi sulle funzioni del circuito per circoscrivere il guasto ad un'area limitata. Poiché occorre aver capito il principio di funzionamento del circuito al fine di ragionare sul problema, verranno discussi i funzionamenti di alcuni circuiti tipici.

ANALISI GENERALE DELLA LOGICA SEQUENZIALE

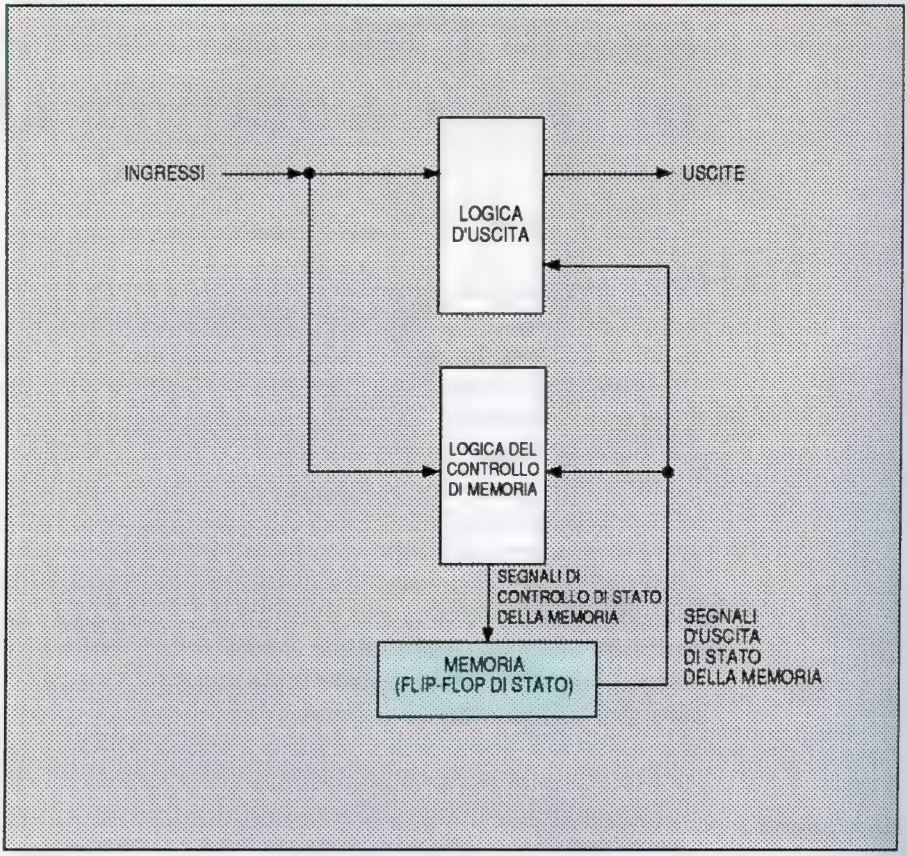
La struttura di base di un sistema a logica sequenziale è mostrata in *figura 5.1*. Le funzioni mostrate in *figura 5.1* vengono divise in due tipi di blocchi logici combinatori (il blocco logico di uscita e il blocco logico di controllo della memoria) e in un blocco di memoria. Lo scopo del blocco logico di uscita è quello di sviluppare i segnali di uscita del sistema in base ai segnali di ingresso e ai segnali presenti sulle uscite degli elementi di memoria.

I segnali di uscita della memoria sono considerati come numeri binari che rappresentano lo stato del sistema. Il numero di stato è l'equivalente decimale del numero binario. Per esempio, se la memoria è formata da tre flip-flop le cui uscite sono 0, 1 e 0 rispettivamente, il sistema viene detto essere in stato 010 o in stato 2 (in quanto 2 è l'equivalente decimale del numero binario 010).

I sistemi logici

Il blocco logico di controllo di memoria è utilizzato per generare i segnali di uscita

Figura 5.1. Tipica struttura di sistema sequenziale


INGRESSI
USCITE
LOGICA D'USCITA
LOGICA DEL CONTROLLO DI MEMORIA
SEGNALI DI CONTROLLO DI STATO DELLA MEMORIA
MEMORIA (FLIP-FLOP DI STATO)
SEGNALI D'USCITA DI STATO DELLA MEMORIA

sequenziali sono sincroni quando vengono sincronizzati da un clock centrale.

di memoria che rappresentano lo stato successivo in base allo stato presente e ai segnali di ingresso. Per esempio, se la macchina è in stato 2 e si suppone che debba passare allo stato 3 in base alle condizioni di ingresso che sono presenti, allora il blocco logico di controllo di memoria genererà dei segnali agli ingressi dei flip-flop di stato che porteranno le loro uscite a diventare 0, 1 e 1. Generalmente, i flip-flop di stato sono di tipo D sincronizzato o flip-flop di tipo J/K; perciò le uscite dei flip-flop (lo stato) cambiano sia sul fronte di salita che sul fronte di discesa del clock del sistema. Questo tipo di sistema viene chiamato sistema sequenziale sincrono perché i cambiamenti di stato procedono di pari passo col clock del sistema (sono cioè sincronizzati). Tutti gli esempi considerati in questo libro e la maggior parte dei sistemi non militari sono sistemi digitali sincroni.

In un sistema reale la parte combinatoria di solito è composta di diverse porte logiche, di decodificatori, di multiplexer, di codificatori e così via; comunque, le memorie ROM e le loro varianti possono essere utilizzate per parte delle funzioni logiche combinatorie. (Un tipo speciale di ROM chiamato rete logica programmabile (PLA) può essere utilizzato anche per svolgere operazioni di logica combinatoria). La memoria del sistema solitamente è data da flip-flop e registri, sebbene possano essere usate memorie ad accesso casuale nei sistemi complessi .

In tutti questi tipi di sistemi il riparatore deve determinare come e sotto che condizioni vengono generate le uscite e attraverso quale sequenza di operazioni passerà la macchina al cambiare degli ingressi. Generalmente ciò richiede che venga tracciata la logica che genera le uscite del sistema e gli ingressi dello stato successivo. Quando viene acquisita un'informazione, questa dovrebbe essere riassunta in un tabulato, in un diagramma di flusso o in una qualsiasi altra forma che possa essere analizzata.

SCHEMA A BLOCCHI DI UN DISTRIBUTORE AUTOMATICO

Un esempio di un sistema a logica sequenziale relativamente semplice è mostrato nello schema a blocchi di *figura 5.2*. La macchina è un semplice distributore automatico a moneta che, per sua stessa natura, è sequenziale. Quando le monete vengono inserite dall'acquirente una dopo l'altra, i loro valori vengono sommati e il valore totale viene memorizzato. Quindi l'acquirente sceglie l'articolo che vuole acquistare. Se il totale delle monete inserite è uguale o maggiore del valore dell'oggetto prescelto, l'articolo verrà consegnato all'acquirente. Se così non è, l'oggetto non verrà emesso e le monete saranno restituite. Se il valore delle monete è maggiore del valore dell'articolo verrà dato un resto. Dopo che le monete sono state restituite o che sono stati emessi l'articolo e il resto, la macchina deve ritornare nella sua condizione di "attesa di ingresso delle monete".

L'unità di controllo si assicura che venga seguita la sequenza operativa descritta. Essa inoltre controlla il segnale di avvenuto inserimento della moneta, quello di scelta dell'articolo e quelli provenienti dal comparatore per determinare quale azione debba essere intrapresa conseguentemente dalla macchina. L'azione potrebbe essere quella di ritornare tutte le monete che sono state inserite, o quella di inviare solo l'articolo scelto, o quella di inviare sia l'articolo scelto che il resto sotto forma di spiccioli.

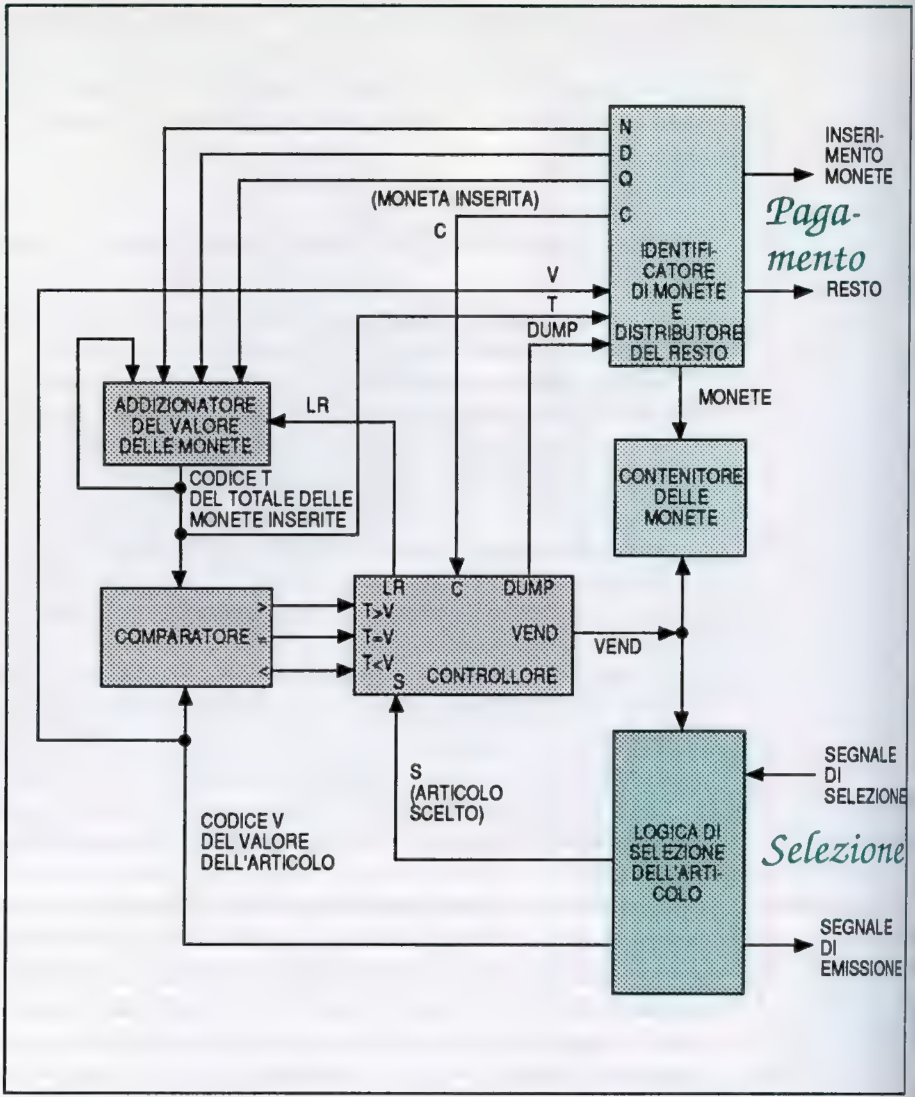
Gli eventi più importanti nell'esempio di un semplice distributore automatico a monete sono: l'identificazione della moneta e il calcolo dopo che una moneta viene inserita, la decisione di restituire le monete o di emettere gli articoli, e l'ammontare del resto che dev'essere restituito.

L'identificatore di monete determina che tipo di moneta è stata inserita e genera un impulso sulla linea di segnale adeguata (N, D o Q) diretto all'addizionatore. Esso inoltre invia un impulso C quando viene inserita una qualsiasi moneta. L'addizionatore dei valori delle monete somma il valore delle monete inserite e memorizza il totale. La logica di selezione dell'articolo determina il valore dell'articolo scelto. I meccanismi elettromeccanici restituiscono tutte le monete se viene ricevuto il segnale DUMP o consegnano l'articolo prescelto all'acquirente se ricevono il segnale VEND. Il meccanismo di controllo del resto restituisce una singola moneta o una combinazione di monete diverse per dare il resto esatto. È necessario comprendere il funzionamento generale o il compito di ogni blocco e del distributore automatico in generale prima di poter analizzare in dettaglio i circuiti che formano ciascun blocco. Fatto questo, è facile determinare la probabile posizione di un guasto nel sistema.

Logica di selezione dell'articolo

Il blocco più semplice di *figura 5.2* è la logica di selezione dell'articolo. Le caratteristiche principali di questa logica sono mostrate sotto forma di circuito nelle *figure 5.3* e *5.4*. La logica di *figura 5.3* ha due funzioni: una è quella di memorizzare l'ingresso di selezione dell'articolo in base ai contatti chiusi di uno dei pulsanti da S1 a

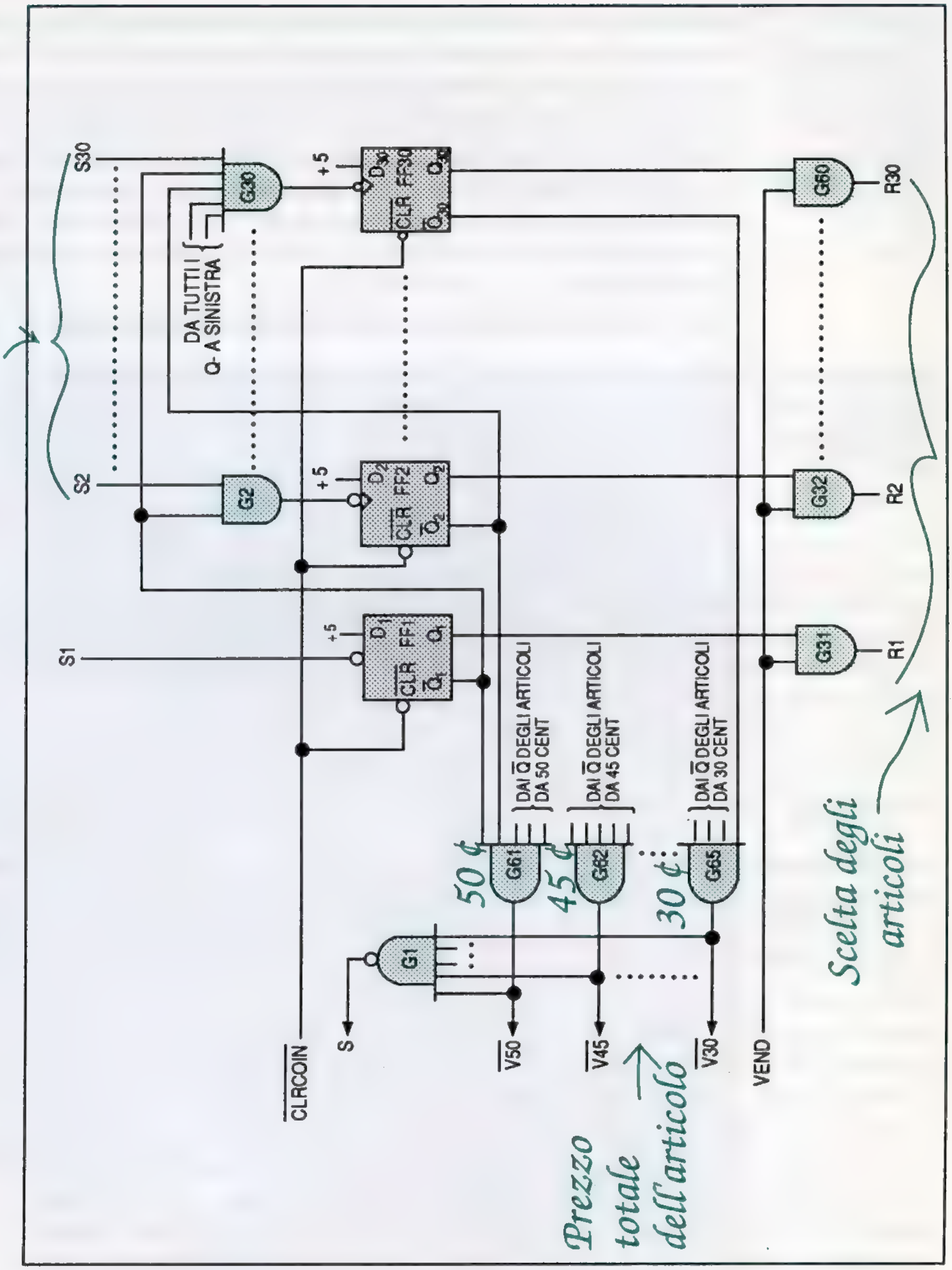
Figura 5.2. Schema a blocchi di un distributore automatico


N
D
Q
C
INSERI-MENTO MONETE
Paga-mento
(MONETA INSERITA)
C
IDENTIFI-CATORE DI MONETE E DISTRIBUTORE DEL RESTO
V
T
DUMP
RESTO
MONETE
ADDIZIONATORE DEL VALORE DELLE MONETE
LR
CODICE T DEL TOTALE DELLE MONETE INSERITE
CONTENITORE DELLE MONETE
COMPARATORE
>
=
<
T>V
T=V
T<V
LR
C
DUMP
VEND
S
CONTROLLORE
VEND
SEGNALE DI SELEZIONE
S (ARTICOLO SCELTO)
CODICE V DEL VALORE DELL'ARTICOLO
LOGICA DI SELEZIONE DELL'ARTI-COLO
Selezione
SEGNALE DI EMISSIONE

Il circuito logico che identifica l'articolo prescelto ha queste funzioni: memorizzare la selezione dell'articolo, determinare il valore dell'articolo, trasmettere i dati.

S30; l'altra è quella di determinare il valore dell'articolo e di passare il relativo segnale al circuito successivo. La logica è organizzata in modo che se il compratore preme più di un pulsante per la scelta dell'articolo viene scelto quello più costoso.
In *figura 5.3*, premendo un singolo pulsante di selezione dell'articolo, S1, il livello di 1 logico all'ingresso D di FF1 apparirà sull'uscita Q_1 mentre uno 0 apparirà sull'uscita $\overline{Q}_1$. Questo livello basso su $\overline{Q}_1$ provocherà l'abbassamento dell'uscita $\overline{V50}$ della porta NAND G1, il che indicherà che è stato prescelto un articolo da 50 cent. Il livello alto di Q_1 abilita l'uscita della porta AND G31 in modo tale che quando si verifica un impulso VEND a livello alto, l'uscita R1 della porta AND G31 andrà al livello alto per emettere l'articolo prescelto. Si noti che quando $\overline{Q}_1$ va al livello basso, esso inibisce le porte dalla G2 alla G36 cosicché nessun altro segnale di commu-

Figura 5.3. Logica di selezione dell'articolo

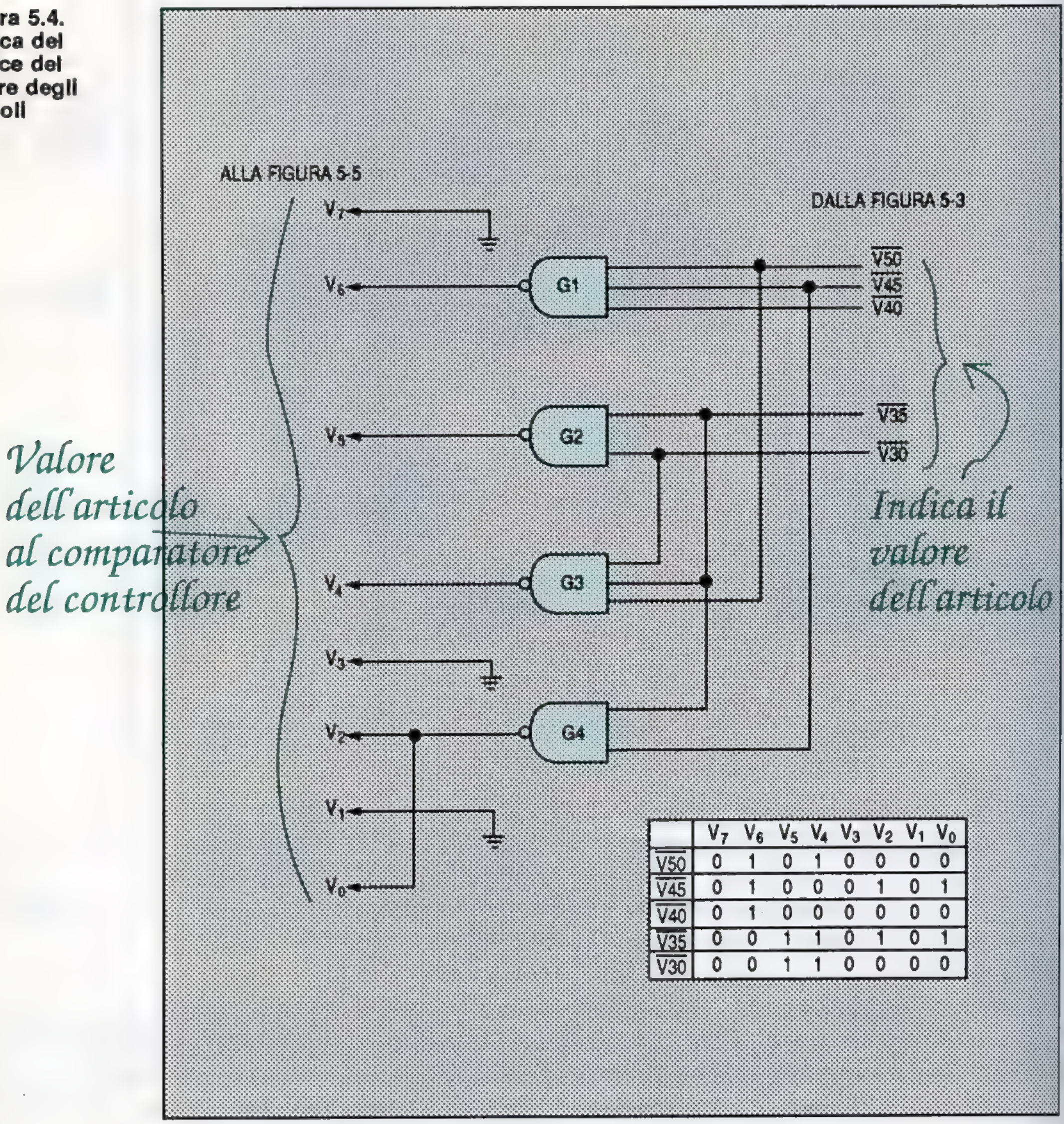

Interruttori di selezione dell'articolo
S30
S2
S1
G30
G2
DA TUTTI I
Q- A SINISTRA
+5
D30
FF30
CLR
Q30
Q̄30
D2
FF2
Q2
Q̄2
D1
FF1
Q1
Q̄1
G50
G32
G31
R30
R2
R1
Scelta degli articoli
DAI Q̄ DEGLI ARTICOLI DA 50 CENT
DAI Q̄ DEGLI ARTICOLI DA 45 CENT
DAI Q̄ DEGLI ARTICOLI DA 30 CENT
50 ¢
45 ¢
30 ¢
G61
G62
G65
G1
CLRCOIN
S
V50
V45
V30
VEND
Prezzo totale dell'articolo

tazione può arrivare ad uno qualsiasi degli altri flip-flop, in modo che venga indicata la selezione del solo articolo 1. In generale, un'uscita 0 sul terminale $\overline{Q}$ di un qualsiasi flip-flop inibirà il segnale del pulsante proveniente da tutti i flip-flop posti alla destra di quello prescelto. Supponendo che gli articoli siano disposti in modo tale che i più costosi siano sulla sinistra e i meno costosi siano sulla destra, questa lo-

gica selezionerà l'articolo più costoso se viene premuto più di un pulsante.
In *figura 5.3* una volta attivato un flip-flop, i segnali da $\overline{V50}$ a $\overline{V30}$ vengono inviati al codice logico del valore dell'articolo mostrato in *figura 5.4*. La logica in *figura 5.4* genera un codice ad 8 bit per indicare il valore dell'articolo prescelto all'unità di controllo in un codice decimale binario (BCD). Il codice BCD rappresenta ciascuna delle cifre decimali come un codice binario a 4 bit. Perciò, se il valore dell'articolo è 35 cent (si verifica un livello basso su $\overline{V35}$), il codice sulle linee da V_7 a V_0 sarà 0011 0101 che sono i codici a 4 bit per 3 e 5. Il lettore può verificare il codice sulle altre linee da V_7 a V_0 per altri valori.
Un qualsiasi livello basso sulle linee da $\overline{V50}$ a $\overline{V30}$ provocherà un livello alto sulle

Un codificatore genera un codice BCD a 8 bit che rappresenta il valore della moneta, ed invia questo dato all'unità di controllo.

Figura 5.4. Logica del codice del valore degli articoli



Valore dell'articolo al comparatore del controllore

Indica il valore dell'articolo

	V_7	V_6	V_5	V_4	V_3	V_2	V_1	V_0
$\overline{V50}$	0	1	0	1	0	0	0	0
$\overline{V45}$	0	1	0	0	0	1	0	1
$\overline{V40}$	0	1	0	0	0	0	0	0
$\overline{V35}$	0	0	1	1	0	1	0	1
$\overline{V30}$	0	0	1	1	0	0	0	0

uscite delle rispettive porte NAND G1-G4. Si noti che le uscite V_1, V_3 e V_7 sono collegate direttamente a massa. Il loro valore è sempre 0 perché un 1 non sarà mai necessario in queste posizioni per i numeri utilizzati. I valori binari su V_0, V_2, V_4, V_5 e V_6 vengono generati dalle porte NAND G1-G4 mostrate in *figura 5.4*. I valori binari per queste uscite sono mostrati nella tabella. Le uscite da V_0 a V_7 vengono inviate al comparatore per il confronto con la somma di denaro inserita.

Addizionatore e comparatore del valore delle monete

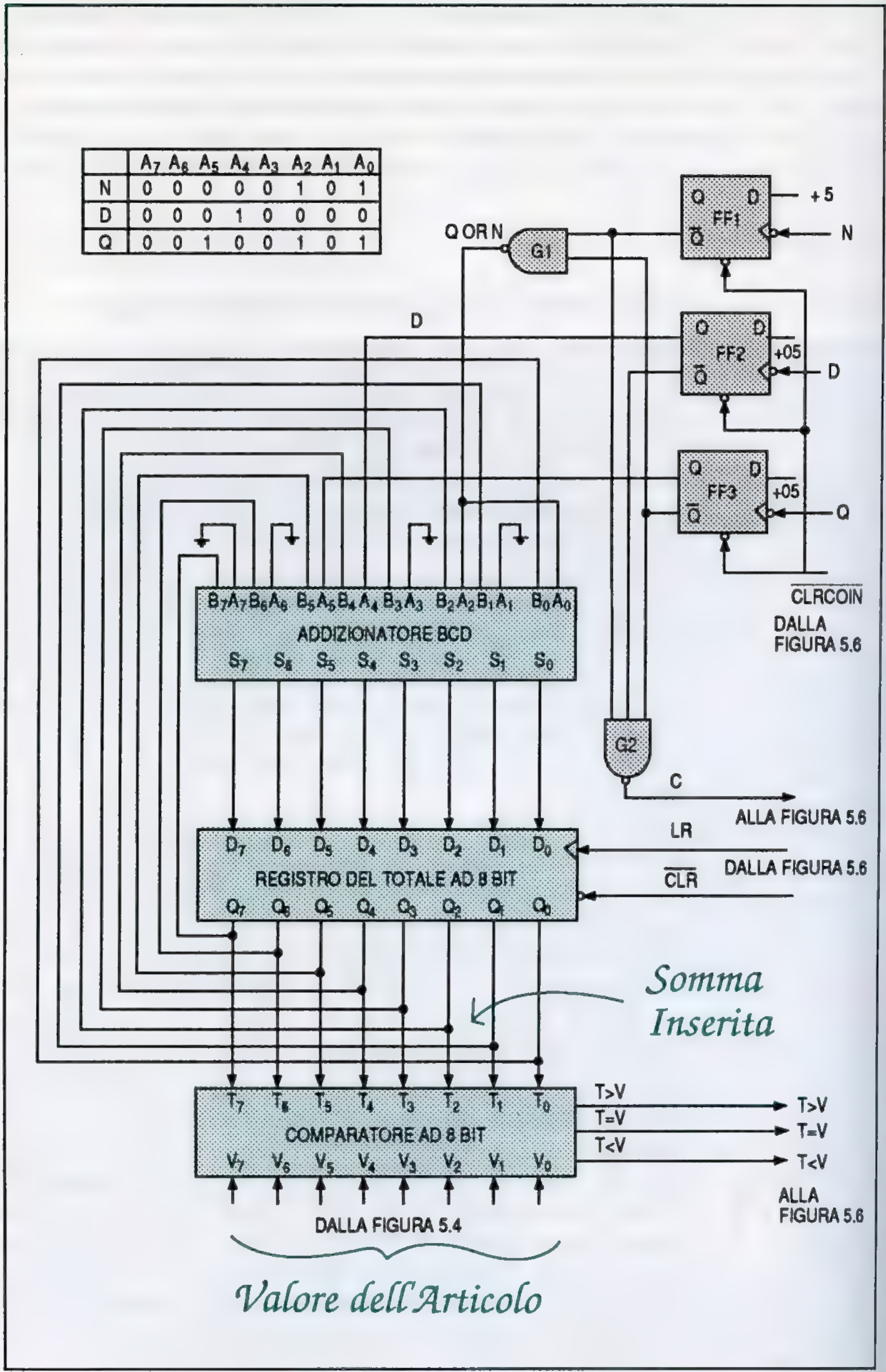
Consideriamo ora l'addizionatore e il comparatore del valore delle monete di *figura 5.2*. Questo circuito è mostrato in *figura 5.5*. Naturalmente l'addizionatore e il comparatore sono mostrati sotto forma di schema a blocchi, ma sono costituiti in realtà da diversi circuiti integrati.

I circuiti logici ed addizionatori determinano il valore delle monete inserite, il quale viene inviato al comparatore. Il comparatore determina la relazione tra il valore delle monete inserite e quello dell'articolo scelto.

In figura sono mostrati più dettagliatamente i circuiti per generare il valore delle monete e il segnale C a partire dai segnali N, D e Q generati dal meccanismo di identificazione delle monete. Un impulso sulla linea N, D o Q generato dal meccanismo di identificazione delle monete provocherà a sua volta il verificarsi di un 1 sull'uscita Q del corrispondente flip-flop. L'uscita $\overline{Q}$ va ad un livello basso, il che provoca il passaggio dell'uscita C di G2 al livello 1. I segnali $\overline{Q}$ di FF1 ed FF3 sono applicati a G1 la cui uscita insieme con l'uscita Q di FF2 generano il codice BCD, mostrato nella tabella, il quale corrisponde al valore delle monete in ingresso, in modo molto simile a quello con cui è stato prodotto il codice di valore dell'articolo nel circuito di *figura 5.4*. Il codice BCD viene applicato agli ingressi dell'addizionatore compresi tra A_7 e A_0.

L'altra serie di impulsi, quelli compresi tra B_7 e B_0, riceve il valore totale dal registro del totale a 8 bit. Le uscite di somma dell'addizionatore BCD (da S_7 a S_0) rappresentano sempre la somma del totale attuale più il valore della nuova moneta inserita. Perciò, se il totale attuale rappresenta 15 cent (S_7 - S_0 = 0001 0101) e viene inserito un quarter (A_7 - A_0 = 0010 0101), le uscite della nuova somma rappresenteranno 40 cent (S_7 - S_0 = 0100 0000). Questo valore viene anche applicato agli ingressi D_7 - D_0 del registro totale. Quando l'impulso LR generato dall'unità di controllo di *figura 5.2* viene inviato al registro del totale, il valore su D_7 - D_0 viene memorizzato nel registro del totale e appare su Q_7 - Q_0. Viene inoltre applicato a T_7 - T_0, che sono gli ingressi del comparatore. L'altra serie di ingressi del comparatore riceve il codice BCD per il valore dell'articolo nel modo che viene determinato nei circuiti di *figura 5.3* e di *figura 5.4*. La funzione del comparatore è quella di indicare la relazione tra la somma di denaro inserita e il valore dell'articolo prescelto. Il comparatore indica ciò portando una delle sue uscite al livello logico 1. Le uscite possibili sono T>V (totale delle monete maggiore del valore dell'articolo), T=V (totale delle monete uguale al valore dell'articolo), oppure T<V (totale delle monete minore del valore dell'articolo). Naturalmente, queste uscite sono significative solo dopo che è stata inserita l'ultima moneta ed è stato prescelto l'articolo: perciò l'unità di controllo ignora questi segnali finché non viene ricevuto il segnale S (segnale di avvenuta scelta dell'articolo in *figura 5.2*).

Figura 5.5. Addizionatore e comparatore del valore delle monete



	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
N	0	0	0	0	0	1	0	1
D	0	0	0	1	0	0	0	0
Q	0	0	1	0	0	1	0	1

Q OR N
G1
FF1
+5
N
D
FF2
+05
D
FF3
+05
Q
$\overline{\text{CLRCOIN}}$
DALLA FIGURA 5.6
$B_7 A_7 B_6 A_6$ $B_5 A_5 B_4 A_4$ $B_3 A_3$ $B_2 A_2 B_1 A_1$ $B_0 A_0$
ADDIZIONATORE BCD
S_7 S_6 S_5 S_4 S_3 S_2 S_1 S_0
G2
C
ALLA FIGURA 5.6
LR
DALLA FIGURA 5.6
$\overline{\text{CLR}}$
D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0
REGISTRO DEL TOTALE AD 8 BIT
Q_7 Q_6 Q_5 Q_4 Q_3 Q_2 Q_1 Q_0
T_7 T_6 T_5 T_4 T_3 T_2 T_1 T_0
COMPARATORE AD 8 BIT
V_7 V_6 V_5 V_4 V_3 V_2 V_1 V_0
T>V
T=V
T<V
ALLA FIGURA 5.6
DALLA FIGURA 5.4

Somma Inserita

Valore dell'Articolo

Unità di controllo

Basandosi sui segnali d'ingresso provenienti dal comparatore, dalla logica di selezione dell'articolo e dall'identificatore di monete, l'unità di controllo fornisce segnali sia per emettere l'articolo scelto e dare il resto, se necessario, che per ritornare le monete inserite.

Finora sono stati esaminati i circuiti di ingresso e di uscita fondamentali. Questi funzionano sotto la direzione dell'unità di controllo del sistema, la quale effettua una certa sequenza di eventi che dipende dagli eventi che sono già avvenuti e dai segnali di ingresso presenti. Perciò il sistema di *figura 5.2* è un sistema a logica sequenziale. Il circuito dell'unità di controllo è mostrato in *figura 5.6* ed è un po' più complicato da analizzare rispetto ai circuiti delle *figure 5.3*, *5.4* e *5.5*. *La chiave per comprendere l'unità di controllo di un sistema sequenziale è determinare la sequenza degli stati e le uscite generate in ogni stato.* Si ricordi che lo *stato attuale* è il codice binario all'uscita dei flip-flop di stato in un particolare istante. Lo *stato successivo* sarà il codice binario attualmente all'ingresso dei flip-flop di stato.

Il particolare progetto di unità di controllo mostrato in *figura 5.6* è un decodificatore di stato. Il codice di stato a 3 bit sull'uscita dei flip-flop di stato viene inviato al decodificatore 3 a 8. Quindi l'uscita del decodificatore che è a 0 indica in che stato si trova l'unità di controllo. Perciò, se le uscite dei flip-flop sono 000, l'uscita 0 del decodificatore sarà bassa e si dirà che l'unità di controllo è nello stato 0 oppure S000. Osservando le uscite dell'unità di controllo generate in base all'uscita del decodificatore e gli ingressi generati ai terminali D dei flip-flop di stato, è possibile determinare che attività vengono svolte nello stato attuale e quale sarà lo stato seguente. *Questi due fatti definiscono completamente cosa fa uno stato e cosa succede in seguito.* Una volta nota questa informazione per tutti gli stati, il comportamento totale della macchina potrà essere compreso e riassunto.

L'approccio più semplice all'analisi di una macchina sequenziale consiste nel cominciare con il primo codice di stato e vedere quali sono le uscite e lo stato successivo. Questo procedimento va poi ripetuto per tutti i codici di stato finché l'analisi sarà completa.

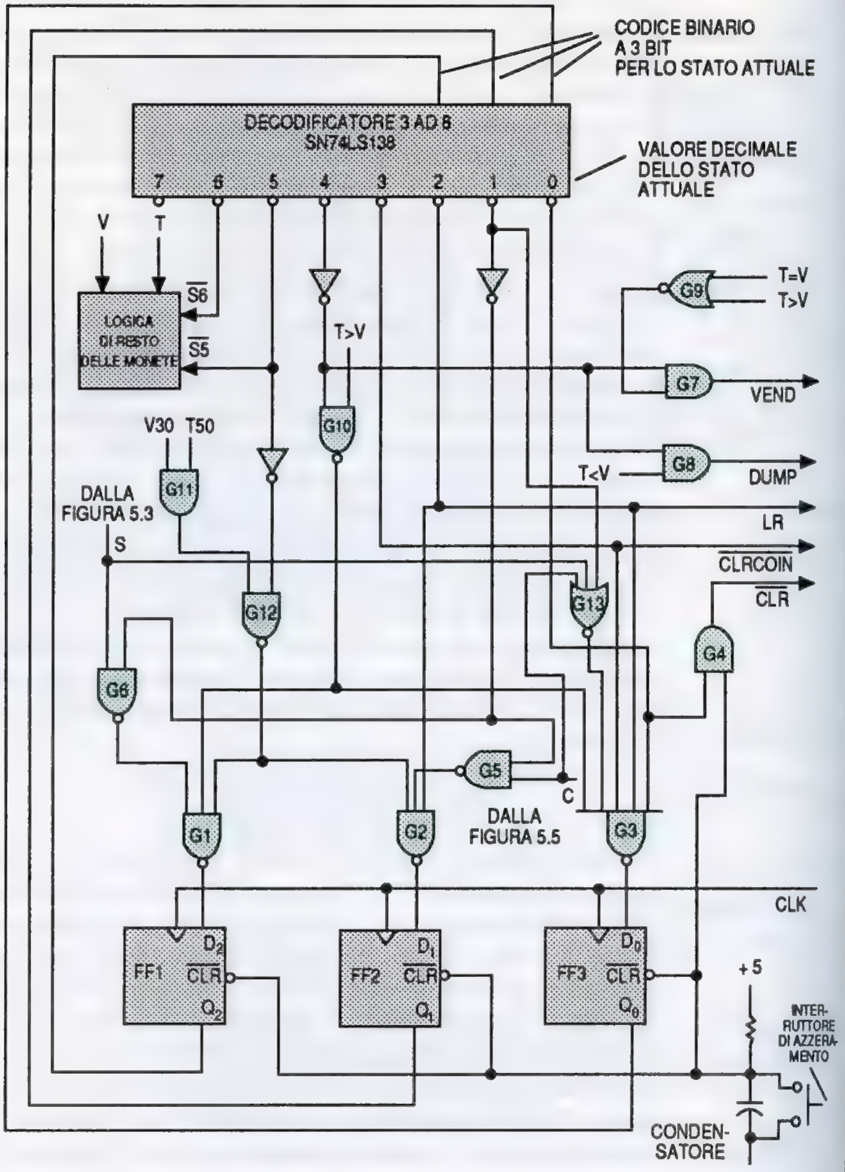
Stato 0

Poiché il circuito è sequenziale ciascuno stato dipende dai segnali d'ingresso presenti e dallo stato in cui il sistema si trova quando arriva il segnale d'ingresso. Lo stato 0 è lo stato iniziale che porta il controllore allo stato 1.

Lo stato 0 si ottiene quando si accende l'alimentatore per la prima volta perché i tre flip-flop di stato FF1, FF2 e FF3 vengono azzerati dall'azione della rete RC collegata con l'alimentatore e con i flip-flop di stato. Questo circuito è collocato nell'angolo basso a destra in *figura 5.6*. Quando l'alimentatore è acceso, la tensione nella giunzione tra il condensatore e il resistore è 0 volt per un breve intervallo di tempo: perciò viene applicato uno 0 logico agli ingressi $\overline{CLR}$ dei flip-flop di stato. Ciò porta Q_0, Q_1 e Q_2 a 000 e determina lo stato 0 attraverso il decodificatore. Dopo poco tempo, il condensatore si carica al livello +5V (1 logico) cosicché viene rimosso l'azzeramento e i flip-flop sono liberi di rispondere ai loro ingressi D. Il ripristino dello stato 0 può essere provocato in ogni istante premendo il pulsante di reset che scarica il condensatore a 0 volt in modo tale da azzerare i tre flip-flop. Dopo che il contatto è stato aperto il condensatore si ricarica di nuovo fino al livello logico 1.

Per determinare quali uscite vengano generate in un determinato stato, si deve seguire la linea di uscita del decodificatore di quello stato verso tutte le destinazioni. Per lo stato 0, l'uscita del decodificatore va alle porte G3 e G4. La porta G4 produce un segnale $\overline{CLR}$ che annulla ogni selezione e azzera i flip-flop delle monete e il registro del totale. Perciò lo stato 0 è uno stato che ripristina la macchina nelle condizioni ini-

Figura 5.6. Logica del controllore


CODICE BINARIO
A 3 BIT
PER LO STATO ATTUALE
DECODIFICATORE 3 AD 8
SN74LS138
7
6
5
4
3
2
1
0
VALORE DECIMALE
DELLO STATO
ATTUALE
V
T
$\overline{S6}$
$\overline{S5}$
LOGICA
DI RESTO
DELLE MONETE
T>V
G9
T=V
T>V
G7
VEND
G10
V30
T50
G8
T<V
DUMP
G11
DALLA
FIGURA 5.3
S
LR
$\overline{CLRCOIN}$
G12
G13
$\overline{CLR}$
G4
G6
G5
C
DALLA
FIGURA 5.5
G1
G2
G3
CLK
FF1
D_2
$\overline{CLR}$
Q_2
FF2
D_1
$\overline{CLR}$
Q_1
FF3
D_0
$\overline{CLR}$
Q_0
+5
INTER-
RUTTORE
DI AZZERA-
MENTO
CONDEN-
SATORE

ziali. Lo stato 0 all'ingresso della porta G3 produce un 1 su D_0. Questo fa sì che lo stato successivo (che si verifica in corrispondenza del fronte di salita del segnale CLK (clock)) sia 001 ovvero lo stato 1.

Stato 1

L'unità di controllo resta nello stato 1 finché non viene ricevuta una moneta per generare un segnale C o finché non arriva un segnale S corrispondente alla selezione di un articolo.

Quando l'unità di controllo è nello stato S001 (stato 1), non viene prodotta alcuna uscita, ma viene applicato un 1 su un ingresso di G5 e G6 per abilitarli. Finché non arriva in ingresso un segnale C oppure S, l'unità di controllo viene mantenuta nello stato 1 dall'uscita del decodificatore attraverso G13. Se si verifica il segnale C, l'uscita 0 di G5 viene applicata a G2 che a sua volta applica un 1 a D_1. Perciò le uscite del flip-flop vanno a 010, cioè allo stato 2. Se si verifica un segnale S, le uscite vanno a 100 o allo stato 4. Quindi, quando si è nello stato 1, lo stato successivo potrebbe essere lo stato 1 (se non si hanno segnali S o C in ingresso), lo stato 2 (se si ha il segnale C in ingresso) o lo stato 4 (se si ha il segnale S in ingresso).

Stati 2 e 3

Quando viene ricevuta una moneta l'unità di controllo va nello stato 2, quindi nello stato 3 e ritorna allo stato 1 in attesa della moneta successiva.

Quando viene inserita una moneta si genera il segnale C, cosicché l'unità di controllo passa allo stato 2 e viene generato il segnale LR per l'acquisizione del totale del valore delle nuove monete inserite nel registro del totale. L'uscita 2 del decodificatore viene inviata anche a G2 e G3 cosicché il codice dello stato successivo è 011. I flip-flop di tipo D ricevono il codice 011 che fa avanzare il decodificatore allo stato 3 alla successiva transizione dal livello basso al livello alto del segnale CLK. Nello stato 3 viene generato il segnale $\overline{\text{CLRCOIN}}$ per azzerare i flip-flop delle monete, in modo tale che siano pronti per l'introduzione di una nuova moneta. L'uscita 3 del decodificatore va anch'essa a G3 in modo tale da provocare un 1 su FF3, cosicché lo stato successivo sia lo stato 1. Allora l'unità di controllo attende nuovamente l'introduzione di un'altra moneta o la selezione di un articolo.

Stato 4

Quando viene fatta una selezione, l'unità di controllo va allo stato 4 per determinare se l'articolo può essere venduto in quanto è stato inserito denaro a sufficienza, o se le monete devono essere restituite.

Quando tutte le monete sono state inserite ed è stata effettuata la selezione, il segnale S fa passare l'unità di controllo dallo stato 1 allo stato 4. Questo è il momento in cui la logica deve determinare se è stato inserito denaro sufficiente per l'articolo prescelto. Dall'uscita del decodificatore allo stato 4 viene applicato un 1 di abilitazione a G7 e a G8. Se è stato inserito denaro sufficiente, come indicato sia da T = V che da T > V, G7 produrrà l'uscita VEND per emettere l'articolo scelto. Invece, se T < V, allora G8 produrrà un'uscita DUMP per restituire le monete inserite. Lo stato successivo viene anch'esso determinato dal confronto fra V e T. Se T < V o T = V, lo stato successivo è 0. Se T > V, lo stato successivo è 5 perché su FF1 ed FF3 è prodotto un 1 dall'uscita di G10.

Stati 5 e 6

L'unità di controllo raggiunge lo stato 5 quando T > V; cioè, quando è stato inserito troppo denaro per l'articolo scelto e deve essere ritornato un resto all'acquirente.

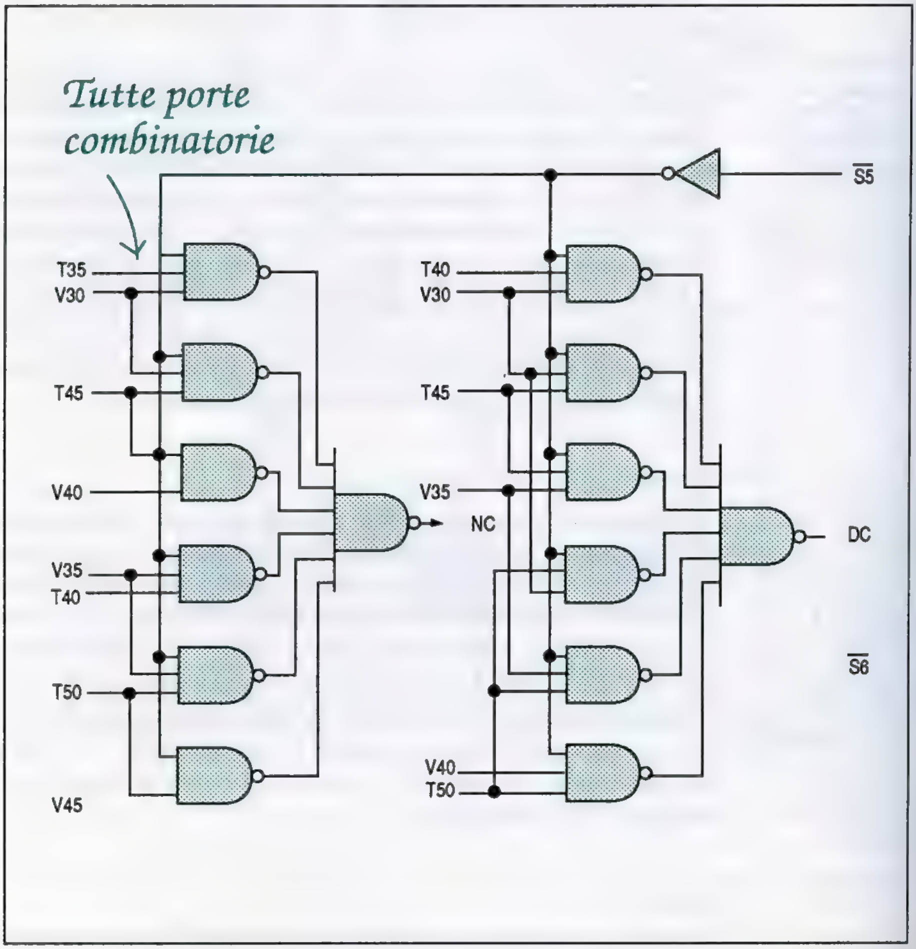
L'unità di controllo entra nello stato 5 e nello stato 6 se deve essere dato il resto. Dopo aver dato il resto ritorna allo stato 0.

L'uscita del decodificatore allo stato 5 abilita il circuito che genera i segnali di controllo del resto ad emettere come resto un nichel, un dime o entrambi. Quindi l'unità di controllo passa allo stato 0 per preparare la selezione successiva, a meno che il valore dell'articolo sia di 30 cent ($\overline{V30}$ è basso) e il totale delle monete inserite sia 50 cent (T50 è alto). In questo caso, l'uscita di G11 va ad un livello alto, il che, unito al fatto che il controllo sia allo stato 5, provoca la trasmissione di un valore basso da G12 agli ingressi di G1 e G2. Ciò produce il codice 110 che fa passare l'unità di controllo allo stato 6, la qual cosa avverte la logica di resto che è necessario restituire un altro dime in resto. A questo punto l'unità di controllo passa allo stato 0.

Circuito di controllo del resto

Il circuito che genera i segnali NC (da nichel change ossia resto in nichel) e i segnali DC (da dime change ossia resto in dime) che azionano il meccanismo di restituzione delle monete è mostrato in *figura 5.7*. Il segnale DC viene generato quando lo sta-

Figura 5.7. Circuito di controllo del resto



to 6 è attivo ($\overline{S6}$) o perché si è verificata una delle 6 differenti combinazioni del valore dell'articolo in funzione del totale delle monete inserite quando è attivo lo stato 5 ($\overline{S5}$). Per esempio, se il valore dell'articolo è di 30 cent e il totale delle monete inserite è 40 cent (V30 in AND con T40), verrà restituito 1 dime. Analogamente, il segnale NC viene generato quando si verifica una qualsiasi delle 6 possibili combinazioni tra valore dell'articolo e valore totale delle monete. Per esempio, se il valore dell'articolo è minore di 5 cent o di 15 cent rispetto al totale delle monete, deve essere generato il segnale NC per restituire un nichel. Il lettore dovrebbe verificare ciascuna delle combinazioni mostrate.

Il circuito di controllo del resto guida il meccanismo del resto in nickel e dime quando lo stato 6 è attivo. Il resto reso si basa sui dati in ingresso durante lo stato 5.

Descrizione della sequenza dell'unità di controllo

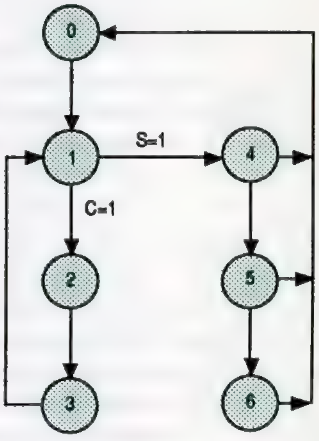
Una volta che i singoli circuiti combinatori di ingresso e di uscita e la sequenzializzazione dell'unità di controllo sono stati capiti, essi possono essere riassunti in una forma tale da costituire un riferimento più semplice. Questo può essere fatto con una lista passo a passo delle operazioni o con un diagramma di flusso. In *figura 5.8* ciò viene mostrato per l'esempio del distributore automatico, il quale viene rappresentato come programma della sequenza di controllo.

Avendo descritto il comportamento della macchina in questa forma, il suo compor-

Figura 5.8. Sintesi del funzionamento di un distributore automatico

Stato	**Stato**	**Azione corrispondente**
0	S000	Azzera i Flip-Flop e il Registro del Totale GOTO S001
1	S001	Se C = 0 e S = O GOTO S001 Se S = 1 allora GOTO S100 Se C = 1 allora GOTO S010
2	S010	Aggiungi il Valore della Moneta al Registro del Totale GOTO S011
3	S011	Azzera i Flip-Flop delle Monete GOTO S001
4	S100	Se T >= V allora VEND Se T < V allora DUMP Se T <= V allora GOTO S000 Altrimenti GOTO S101
5	S101	Emetti il Resto Se (V30 e C50) = 0 allora GOTO S000 Altrimenti GOTO S110
6	S110	Emetti il Resto in Dime GOTO S000

a. Programma della Sequenza di Controllo



b. Diagramma

tamento generale può essere facilmente capito e tenuto come riferimento. Inoltre, è facile vedere l'effetto di un dato guasto sulla sequenza delle operazioni. È inoltre possibile determinare l'effetto di un guasto sul comportamento di un circuito combinatorio utilizzando le stesse tecniche descritte negli ultimi due capitoli. Alcuni guasti tipici e il loro effetto sul funzionamento del distributore automatico verranno considerati in seguito.

GUASTI NEI SISTEMI SEQUENZIALI

I sistemi sequenziali mostrano lo stesso tipo di guasti dei sistemi combinatori. La miglior tecnica di ricerca dei guasti consiste nel ragionamento basato sulle funzioni del sistema per localizzare i guasti, i quali verranno poi esattamente individuati con l'aiuto della strumentazione di test.

Anche nei sistemi sequenziali si verificano gli stessi corto circuiti, circuiti aperti e guasti di componenti che si trovano nei sistemi combinatori. Molti dei procedimenti utilizzati per isolare i guasti nei sistemi a logica combinatoria si applicano anche ai sistemi sequenziali. Chiaramente il processo di ragionamento utilizzato per circoscrivere il guasto ad una certa unità funzionale nel sistema è un procedimento che viene utilizzato in tutti i tentativi di ricerca ed eliminazione dei guasti. Quindi, misurazioni effettuate con strumenti di test aiutano a localizzare il problema che sta causando il malfunzionamento del sistema.

Guasti nei dispositivi di memoria

I dispositivi di memoria sono propri solo dei sistemi sequenziali e consistono di flip-flop e registri; entrambi funzionano nello stesso modo. Il più comune guasto di un flip-flop di tipo D è un'uscita che non risponde ad un ingresso.

I soli tipi di elementi che si trovano nell'esempio utilizzato in questo capitolo e che non erano stati usati nei sistemi a logica combinatoria sono i dispositivi di memoria. Tra questi sono inclusi i flip-flop e il registro di memorizzazione del totale delle monete. Questi rappresentano fondamentalmente lo stesso tipo di componente, in quanto il registro è semplicemente un insieme di 8 flip-flop che hanno in comune la stessa linea di clock. Perciò il solo nuovo tipo di componente che deve essere considerato dal punto di vista della ricerca dei guasti è il flip-flop di tipo D. Questo flip-flop ha un ingresso D, due uscite Q e $\overline{Q}$ (complementari una all'altra), un ingresso di clock e un ingresso di azzeramento. Possono anche esserci altri ingressi di controllo, ma, per l'esempio considerato in questo capitolo, sarà sufficiente considerare solo questi ingressi e uscite fondamentali.

Il guasto più comune che potrebbe essere osservato in un flip-flop di tipo D è che la sua uscita non risponda agli ingressi. In altre parole, che l'uscita Q sia fissa o al livello 0 o al livello 1. L'altra uscita $\overline{Q}$ sarebbe fissa al livello opposto a meno che ci fosse un guasto nel circuito di uscita del dispositivo. Sia i guasti che comportano il blocco a 0 che quelli che comportano il blocco a 1 in un flip-flop possono avere effetti disastrosi sul funzionamento di un sistema sequenziale, particolarmente se il flip-flop guasto è uno dei flip-flop di stato.

Blocco a 0

Un caso di guasto dell'uscita è la condizione di blocco a 0, nella quale Q rimane a 0 e $\overline{Q}$ rimane ad 1 indipendentemente dalle condizioni di ingresso. Se viene applicato un 1 all'ingresso D e un impulso di clock all'ingresso di clock o di acquisizione e Q rimane comunque a 0, allora o l'ingresso $\overline{CLR}$ ha subito un corto a massa ed è in condizione di azzeramento permanente o c'è un'apertura sulle linee del clock o di ingresso D. Questi guasti possono essere interni al flip-flop o nel cablaggio esterno. Provando il componente con l'isolamento di piedini o sostituendolo con un componente sicuramente funzionante si potrà escludere la possibilità di un guasto interno. I guasti nel cablaggio possono essere determinati con le stesse tecniche di rilevazione del segnale e di misurazione con il voltmetro discusse nei capitoli precedenti.

Blocco a 1

Se il flip-flop non si azzera quando viene applicato un livello basso al terminale $\overline{\text{CLR}}$, allora c'è un guasto interno nel circuito di azzeramento oppure nel cablaggio (probabilmente un circuito aperto). Se il flip-flop è bloccato al livello 1 (Q è permanentemente 1 e $\overline{\text{Q}}$ è permenentemente 0) indipendentemente dallo 0 applicato all'ingresso D e dall'applicazione di un impulso di clock adatto, deve trattarsi ancora di un guasto interno o di un guasto (probabilmente un circuito aperto) sulla linea D o sulla linea del clock. Tali problemi possono essere individuati ed eliminati con gli stessi procedimenti utilizzati per ovviare alla condizione di blocco a 0.

Analisi di guasti in un sistema sequenziale

In un sistema sequenziale guasto il riparatore osserverà che qualche comportamento del sistema non è normale e, attraverso il ragionamento, determinerà la causa più probabile di tale comportamento. Quindi egli dovrà utilizzare degli strumenti di test per individuare la causa precisa. Questo sarà il tipo di approccio che verrà considerato nei prossimi paragrafi. Naturalmente, non tutti i problemi che possono verificarsi in un dato sistema sequenziale possono essere esaminati in uno spazio limitato come quello che è qui disponibile, ma possono essere trattati esaurientemente quelli più significativi. Il sistema del distributore automatico trattato nella sua completezza dalle *figure 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8* servirà come sistema esemplificativo in quanto il suo funzionamento è già stato analizzato dettagliatamente.

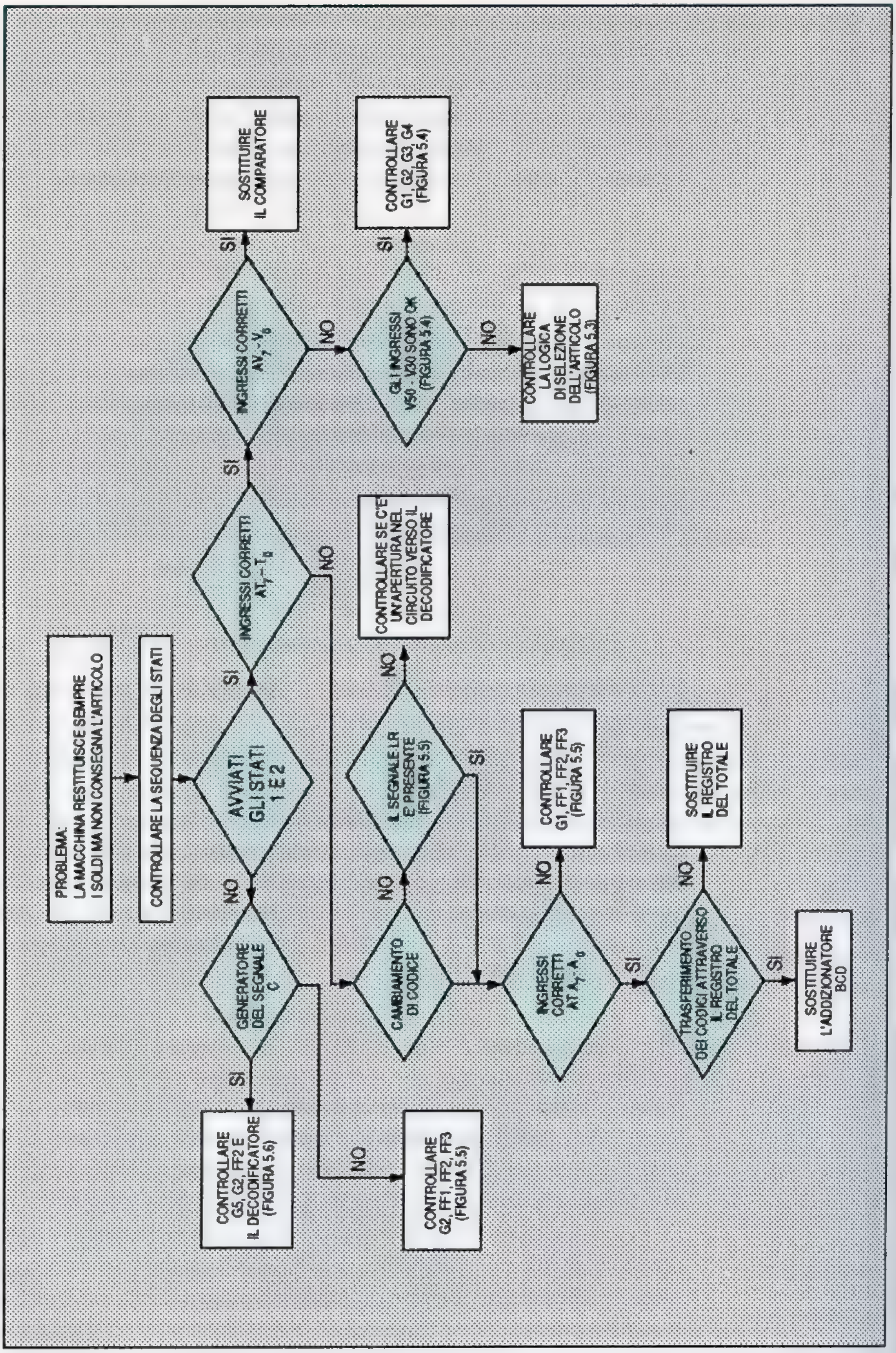
Sintomo 1

Se la macchina restituisce le monete inserite ma non emette l'articolo, il miglior punto di partenza è un controllo per verificare se i flip-flop vanno negli stati 2 e 3 quando vengono inserite le monete.

Un malfunzionamento della macchina che infastidisce non poco l'acquirente è quello per cui la macchina accetta le monete inserite e non consegna l'articolo scelto, ma ritorna le monete. Si faccia riferimento alla *figura 5.8* e si esaminino le operazioni. Dato che le monete vengono restituite, la macchina apparentemente raggiunge lo stato 4 per generare il segnale DUMP, ma non genera mai il segnale VEND. Questo comportamento può verificarsi solo se la macchina, quando raggiunge lo stato 4, determina sempre che il totale delle monete inserite è minore del valore dell'articolo. A questo punto, la macchina rende le monete e ritorna allo stato 0 e quindi allo stato 1.

Si noti in *figura 5.8* che lo stato 4 può essere raggiunto direttamente dallo stato 1 senza mai passare per gli stati 2 e 3 nei quali i valori delle monete vengono sommati. Se ciò accade, allora T sarà sempre 0 e si verificherà quindi il guasto osservato. Perciò,

Figura 5.9. Diagramma di flusso per la ricerca dei guasti


PROBLEMA:
LA MACCHINA RESTITUISCE SEMPRE I SOLDI MA NON CONSEGNA L'ARTICOLO
CONTROLLARE LA SEQUENZA DEGLI STATI
AVVIATI GLI STATI 1 E 2
SI
NO
GENERATORE DEL SEGNALE C
SI
CONTROLLARE G5, G2, FF2 E IL DECODIFICATORE (FIGURA 5.6)
NO
CONTROLLARE G2, FF1, FF2, FF3 (FIGURA 5.5)
INGRESSI CORRETTI AT7–T0
SI
NO
INGRESSI CORRETTI AV7–V0
SI
SOSTITUIRE IL COMPARATORE
NO
GLI INGRESSI V50...V30 SONO OK (FIGURA 5.4)
SI
CONTROLLARE G1, G2, G3, G4 (FIGURA 5.4)
NO
CONTROLLARE LA LOGICA DI SELEZIONE DELL'ARTICOLO (FIGURA 5.3)
CAMBIAMENTO DI CODICE
NO
IL SEGNALE LR È PRESENTE (FIGURA 5.5)
NO
CONTROLLARE SE C'È UN'APERTURA NEL CIRCUITO VERSO IL DECODIFICATORE
SI
INGRESSI CORRETTI AT A7–A0
NO
CONTROLLARE G1, FF1, FF2, FF3 (FIGURA 5.5)
SI
TRASFERIMENTO DEI CODICI ATTRAVERSO IL REGISTRO DEL TOTALE
NO
SOSTITUIRE IL REGISTRO DEL TOTALE
SI
SOSTITUIRE L'ADDIZIONATORE BCD

un buon punto da cui partire per individuare i guasti è quello di controllare che i flip-flop di stato passino allo stato 2 e 3 ogni volta che viene inserita una moneta. Se non è così, si controlli che venga prodotto il segnale C (*figure 5.5* e *5.6*) all'ingresso di ogni moneta. Se questo non accade, allora il problema è circoscritto a G2, FF1, FF2 o FF3 in *figura 5.5*. Se C viene generato, allora il problema è circoscritto a G5, G2, FF2 o al decodificatore in *figura 5.6*. La *figura 5.9* riassume quanto detto e la seguente sequenza di ricerca dei guasti.

Se l'unità di controllo passa ciclicamente dallo stato 1 al 2 al 3 e quindi ancora all'1, allora il problema potrebbe trovarsi nel circuito di identificazione e di addizione del valore delle monete, nel circuito di valore dell'articolo o nel comparatore. Sia che il problema osservato si verifichi indipendentemente dalla combinazione di monete inserite o dalla selezione dell'articolo, sia che si verifichi solo per determinate combinazioni di monete inserite e di articolo scelto, è utile determinare le operazioni successive che il riparatore deve compiere. Si supponga che il problema si verifichi indipendentemente dalle combinazioni.

Poiché il comparatore è situato al centro dell'area sospettata, potrebbe essere una buona idea iniziare da qui la ricerca dei guasti. Si controlli il codice binario agli ingressi T_7 - T_0 mentre vengono inserite due o tre monete. Se il codice cambia correttamente per mostrare il totale accumulato, allora i circuiti di identificazione e di addizione del valore delle monete funzionano correttamente. Se il codice non cambia assolutamente, il segnale LR generato dall'uscita del decodificatore nello stato 2 potrebbe non raggiungere il registro del totale o l'ingresso LR stesso potrebbe essere difettoso. Se il codice cambia, ma non in modo corretto, si controllino gli ingressi A_7 - A_0 mentre vengono inserite le monete. Se questi codici sono corretti per ciascuna moneta, come mostrato nella tabella, allora probabilmente l'addizionatore BCD è difettoso, ma si potrebbe sospettare anche del registro del totale. Se il codice all'ingresso del registro del totale viene trasferito all'uscita quando si verifica il segnale LR, allora il problema è nell'addizionatore BCD. Se gli ingressi A_7 - A_0 non sono corretti, il problema è circoscritto a G1, FF1, FF2 o FF3 (*figura 5.5*).

Se gli ingressi del comparatore T_7 - T_0 fossero corretti, allora si dovrebbero controllare gli ingressi del comparatore V_7 - V_0 mentre vengono inserite delle monete e viene selezionato un articolo. Se il codice BCD è corretto per la selezione di un qualsiasi articolo (si veda la tabella in *figura 5.4*), allora deve essere difettoso il comparatore. Se il codice BCD non è corretto, allora deve essere difettosa o la logica del valore dell'articolo o la logica di selezione dell'articolo (*figure 5.3* e *5.4*). Si ritorni quindi ai segnali da V30 a V50 e alle porte di selezione dell'articolo per isolare il problema.

Sintomo 2

Quando la macchina va alternativamente dallo stato 4 (generazione del segnale VEND)

Un problema nel comportamento della macchina che fa certo piacere all'acquirente è quello per cui la macchina continua ad emettere l'articolo e monete di resto finché è vuota. Si faccia riferimento nuovamente alla *figura 5.8*. Questo problema potrebbe verificarsi solo se la macchina oscillasse tra lo stato 4, in cui è generato il segnale VEND, e lo stato 5, nel quale viene emesso il resto. Il comportamento descritallo stato 5 (emissione di resto) la macchina emetterà tutte le monete finché non sarà vuota; il guasto più probabile è un'apertura sul collegamento G12-G2

to potrebbe accadere se la macchina considerasse come stato successivo al 5 lo stato 4 piuttosto che lo stato 0 o 6 come dovrebbe essere correttamente. Questo potrebbe verificarsi se fosse guasto il circuito che genera l'1 sul flip-flop di stato FF2 (*figura 5.6*) quando è attivo lo stato 5. L'unico guasto che potrebbe causare questo problema è un'apertura sulla linea da G12 a G2 (*figura 5.6*). Questo guasto farebbe diventare lo stato 4 successivo allo stato 5 e la macchina oscillerebbe tra gli stati 4 e 5.

Sintomo 3

Un guasto nella porta G7, nella porta G9 o nella linea d'alimentazione comune dei solenoidi potrà essere causa del fatto che la macchina trattenga il denaro ma non emetta né il resto né l'articolo scelto.

Il problema della macchina che fa più arrabbiare l'acquirente è quello per cui la macchina accetta il denaro, ma non emette l'articolo scelto e non ritorna le monete inserite. A volte può darsi che essa dia un resto in base alle monete inserite e al valore dell'articolo scelto. Se la macchina dà un resto, essa raggiunge lo stato 5 di *figura 5.8.* Il segnale VEND può non essere generato a causa di un guasto in G7 o in G9 (*figura 5.6*) o a causa di un'apertura tra la linea VEND dell'unità di controllo e le porte G31-G60 (*figura 5.3*) che trasmettono il segnale di emissione alle bobine elettriche del meccanismo di distribuzione. Naturalmente, un'altra possibile causa non dovuta alla logica è la linea di alimentazione comune alle bobine di distribuzione.

Se non viene distribuito un resto, la macchina potrebbe non essere in grado di raggiungere lo stato 4 in quanto il segnale S non viene generato correttamente (*figura 5.3*) perché G1 è guasta. Se il problema si verifica solo per alcuni articoli, ma non per tutti, allora una delle porte NAND o uno dei flip-flop nella logica di selezione dell'articolo potrebbe essere guasto. Se il segnale S viene generato, allora il guasto potrebbe essere in G6, in G1 o in FF1 (*figura 5.6*).

Quando si verifica un guasto che non permette al sistema di raggiungere lo stato 4 non vengono restituite le monete e non viene emesso alcun articolo.

La macchina potrebbe essere bloccata nello stato 0 a causa di diverse ragioni. FF3 (*figura 5.6*) potrebbe essere bloccato a 0 o perché FF3 è guasto o perché è sempre presente un segnale $\overline{\text{CLR}}$. Un segnale $\overline{\text{CLR}}$ costante potrebbe essere causato da un corto circuito a massa sul circuito stampato, in G4, nel condensatore o nel pulsante di reset. Il decodificatore potrebbe essere guasto oppure potebbe esserci un'apertura sulla linea tra l'uscita 0 del decodificatore e G3, oppure potrebbe essere guasta G3. Uno qualsiasi di questo guasti farebbe restare la macchina nello stato 0.

La macchina potrebbe rimanere bloccata all'interno della sequenza di stato 1, 2 e 3 se il segnale C rimane fisso ad 1 per colpa dei flip-flop di identificazione delle monete FF1, FF2 ed FF3 di *figura 5.5* i quali non sono stati azzerati dal segnale $\overline{\text{CLRCOIN}}$ prodotto dallo stato 3 in *figura 5.6*.. Questo potrebbe accadere se ci fosse un'apertura sulla linea $\overline{\text{CLRCOIN}}$. In realtà, se fossero presenti sia il segnale C che il segnale S a causa di questa condizione, la macchina andrebbe allo stato 7, il quale non è uno stato operativo e il cui stato successivo è lo stato 0. Perciò, l'effetto finale provocato dal segnale C bloccato a 1 è che lo stato 4 non viene mai raggiunto; quindi non vengono restituite monete e l'articolo non viene emesso.

Potrebbero venire inserite monete finché il loro contenitore sia pieno senza che si abbia alcuna risposta dalla macchina. Si noti che la logica è predisposta per tutelare il venditore piuttosto che l'acquirente. Un solo ed improbabile guasto può causare una situazione di "bottino" da parte dell'acquirente, ma molti dei diversi guasti autorizzeranno la macchina a trattenere il denaro dell'acquirente senza vendergli l'articolo.

Porte guaste nel circuito di controllo del resto, un guasto del comparatore oppure un guasto del decodificatore possono causare un errore della macchina nel restituire il resto. Il sistema non raggiunge mai lo stato 5 oppure la macchina non ha più denaro da restituire.

Sintomo 4

Come ultimo esempio dei problemi che possono verificarsi in un distributore automatico, si supponga che la macchina accetti le monete ed emetta l'articolo scelto, ma che non ritorni alcun resto. Naturalmente, potrebbe soltanto trattarsi del semplice caso in cui il distributore del resto abbia finito le monete. Comunque, il guasto potrebbe essere anche dovuto alla macchina che non raggiunge mai lo stato 5 o alle porte del resto (*figura 5.7*) che non ricevono mai il segnale abilitante $\overline{S5}$ dallo stato 5. Una o più porte nel circuito di controllo del resto potrebbero essere guaste; comunque, questo probabilmente non sempre tratterrebbe il resto sotto qualsiasi condizione. La macchina potrebbe passare sempre dallo stato 4 allo stato 0 se l'uscita di G10 (*figura 5.6*) fosse bloccata ad 1 a causa o di un guasto in G10, o di un guasto sulla linea T > V, o di un guasto sulla linea tra l'uscita 4 del decodificatore e G10, o di un guasto nello stesso decodificatore. Uno qualsiasi di questi guasti impedirebbe l'emissione di un resto.

Esistono molti altri malfunzionamenti che possono verificarsi in un distributore automatico. Il lettore potrebbe supporre qualche guasto nel cablaggio o in un componente e quindi cercare di determinare come si comporterebbe la macchina in quella specifica condizione.

Guasti nei dispositivi sequenziali MSI

Il contatore è un dispositivo simile al registro e viene utilizzato per circuiti di temporizzazione e di clock nel progetto dei sistemi sequenziali. I guasti nei circuiti di temporizzazione possono portare confusione nel funzionamento di un sistema sequenziale. Alcuni di questi guasti verranno considerati nei seguenti paragrafi utilizzando come esempio il circuito di generazione di clock per un distributore automatico.

Oscillatore

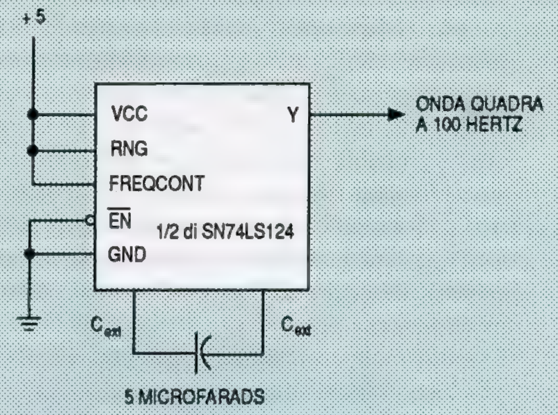
In *figura 5.10* viene mostrato un possibile circuito di generazione del clock. Esso è progettato in modo tale che possa produrre una frequenza di 100 hertz, che ha un periodo di 10 millisecondi. Il principio di funzionamento del dispositivo di *figura 5.10a* è abbastanza semplice. Il valore del condensatore fissa la frequenza dell'onda quadra generata internamente in base alla seguente relazione approssimata:

$$F_o = 500/C$$

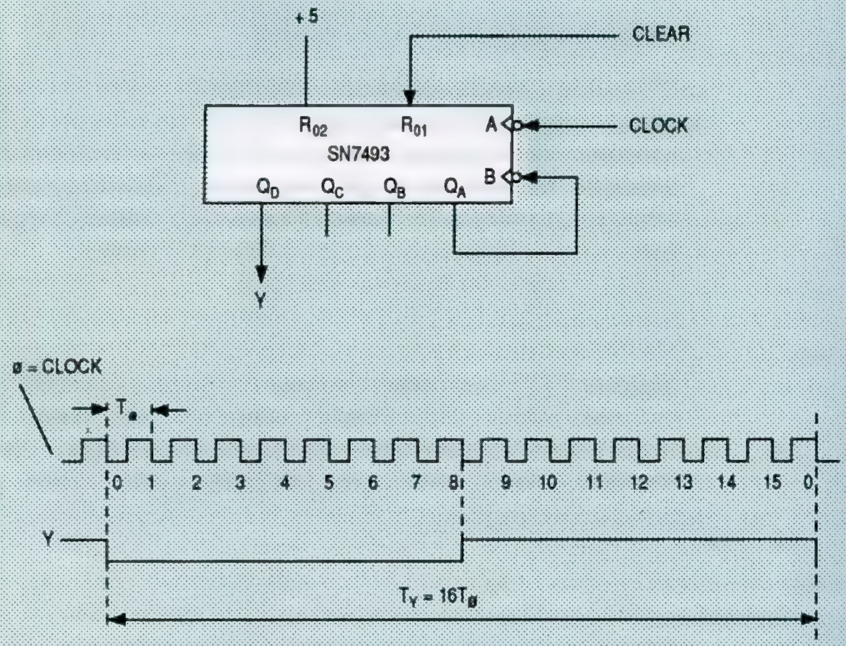
dove la frequenza F_o è in megahertz (milioni di hertz) e C è in picofarad (un milionesimo di microfarad). (Questo particolare dispositivo può produrre una frequenza in uscita fino a un massimo di 40 MHz.) Perciò con un condensatore il cui valore è di 5 microfarad (5 milioni di picofarad) come mostrato, la frequenza è fissata a 500/5.000.000, ossia 100 hertz.

Il terminale $\overline{EN}$ abilita l'onda quadra in uscita quando riceve un livello basso. In questa applicazione, esso è sempre abilitato essendo collegato a massa. I terminali RNG e FREQCONT permettono ad una tensione esterna di controllare la frequenza del-

Figura 5.10. Oscillatori e contatori


+5
VCC
RNG
FREQCONT
EN
GND
Y
1/2 di SN74LS124
ONDA QUADRA
A 100 HERTZ
C ext
C ext
5 MICROFARADS

a. Generatore di Clock


+5
CLEAR
R02
R01
SN7493
A
B
CLOCK
QD
QC
QB
QA
Y
ø = CLOCK
Tø
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 0
Y
TY = 16Tø

b. Contatore Modulo 16

l'uscita in modo che resti attorno al valore fissato dal condensatore. Comunque, in questa applicazione, questi ingressi sono fissati a +5 volt e non hanno un ulteriore controllo sulla frequenza dell'uscita.

Guasti nei circuiti di tempificazione e di clock possono provocare un funzionamento confuso del sistema sequenziale. I problemi più ovvi sono i generatori di clock e i circuiti del contatore guasti.

I problemi più evidenti che possono verificarsi con questo tipo di circuito sono l'assenza di segnale in uscita o una frequenza errata. Se non si ha alcuna uscita la causa sarà probabilmente un circuito integrato difettoso o un circuito aperto nei collegamenti con V_{cc}, GND o $\overline{EN}$. Un'apertura nei collegamenti coi terminali RNG e FREQCONT potrebbe causare una frequenza di valore diverso da quello voluto, ma il circuito produrrebbe comunque un'onda quadra approssimativamente della frequenza esatta. Se il collegamento del terminale FREQCONT avesse subito un corto circuito a massa, ne risulterebbe una frequenza molto bassa. Un'apertura nei collegamenti col condensatore oppure il condensatore stesso potrebbero causare un aumento della frequenza fino ad un massimo di circa 40 megahertz. Se non si scoprono guasti nel cablaggio, si deve concludere che il circuito è difettoso e va sostituito.

Contatore

Il circuito mostrato in *figura 5.10b* è un semplice contatore binario a 4 bit collegato in modo tale da poter dividere per 16 il segnale di ingresso. Il dispositivo mostrato ha 2 ingressi di clock (A e B), 4 uscite Q_A (bit meno significativo), Q_B, Q_C e Q_D (bit più significativo), e 2 linee di controllo di azzeramento R_{01} e R_{02}. Sia R_{01} che R_{02} devono essere ad un livello alto per azzerare il contatore. Dato che R_{02} è collegato in modo da essere sempre al livello alto, R_{01} controlla la funzione di azzeramento. Se R_{01} è al livello basso il conteggio verrà aumentato di 1 ogniqualvolta il segnale A di clock del sistema subisce una transizione da un livello alto a un livello basso. Per questo dispositivo la frequenza di clock massima è tipicamente di 42 megahertz. L'ingresso di clock B è collegato con l'uscita Q_A per fornire la funzione di divisione per 16. L'uscita del contatore che interessa è Q_D e viene detta uscita Y.

L'uscita Y è mostrata in relazione con l'ingresso di clock A sotto forma d'onda. Quando si sono avuti 8 fronti di discesa del segnale A il segnale Y cambierà nel suo complementare (passerà cioè da 0 a 1 o viceversa). Perciò l'uscita Y è un'onda quadra che ha un periodo uguale a 16 volte il periodo del clock di ingresso. Dato che la frequenza è il reciproco del periodo, la frequenza dell'onda quadra Y è 1/16 della frequenza del clock di ingresso. Ecco perché il contatore è detto anche divisore per 16 o contatore modulo 16.

Anche se il modo in cui funzionano i dispositivi LSI e VSLI è fondamentalmente differente, la ricerca dei guasti nei componenti sequenziali comporta le stesse tecniche utilizzate nella ricerca dei guasti

Se il contatore di *figura 5.10b* sbaglia a contare significa che o la linea R_{01} è bloccata al livello 1 (probabilmente per un'apertura nel cablaggio), o che c'è un'apertura sul piedino dell'ingresso A, o che il segnale di ingresso A è bloccato a 0 oppure ad 1 (non ci sono transizioni da un livello all'altro), o che i piedini dell'alimentatore e/o i piedini di massa sono aperti. Se cambia solamente la linea Q_A e le altre uscite sono bloccate a 0 o ad 1, allora si tratta probabilmente di un'apertura sulla linea da Q_A a B o di un corto circuito a massa sulla stessa linea. Altrimenti è possibile che sia difettoso il circuito integrato.

Si può così vedere che il corretto funzionamento dei componenti sequenziali come il temporizzatore, il contatore e il generatore di onde quadre (oscillatore) dipende soprattutto dal fatto che i componenti funzionino e dal fatto che ci siano dei buoni colnei componenti combinatori.

legamenti con i terminali dei componenti, proprio come nel caso dei componenti combinatori. Perciò tali componenti non differiscono di per sé molto da un dispositivo combinatorio per quanto riguarda la ricerca dei guasti. Il riparatore deve determinare se il componente funziona, nel qual caso dovrà essere cercato il guasto nel cablaggio. In alternativa, se il componente è saldato sul circuito stampato, si potrà determinare il buon funzionamento del cablaggio utilizzando le tecniche di rilevazione del segnale e di isolamento dei piedini prima di cercare di rimuovere il componente e di sostituirlo. La differenza tra i circuiti sequenziali MSI ed LSI sta nel loro comportamento di base e nell'analisi del funzionamento del loro circuito come si è potuto vedere nell'esempio del distributore automatico. Le differenze sono molto più significative tra i dispositivi LSI e VLSI (integrati su scala molto grande) che vengono utilizzati nei microcomputer. Molta della rimanente parte del libro riguarderà questa classe così importante di sistemi e la loro apparecchiatura di supporto, a cominciare da una panoramica generale sul loro funzionamento che verrà trattata nei prossimi paragrafi.

INTRODUZIONE AI MICROCALCOLATORI

L'analisi e l'individuazione di un guasto nei microcomputer richiedono che il riparatore conosca le interconnessioni fra i componenti, lavori con molti cambiamenti di stato simultanei dei segnali e capisca come il software opera.

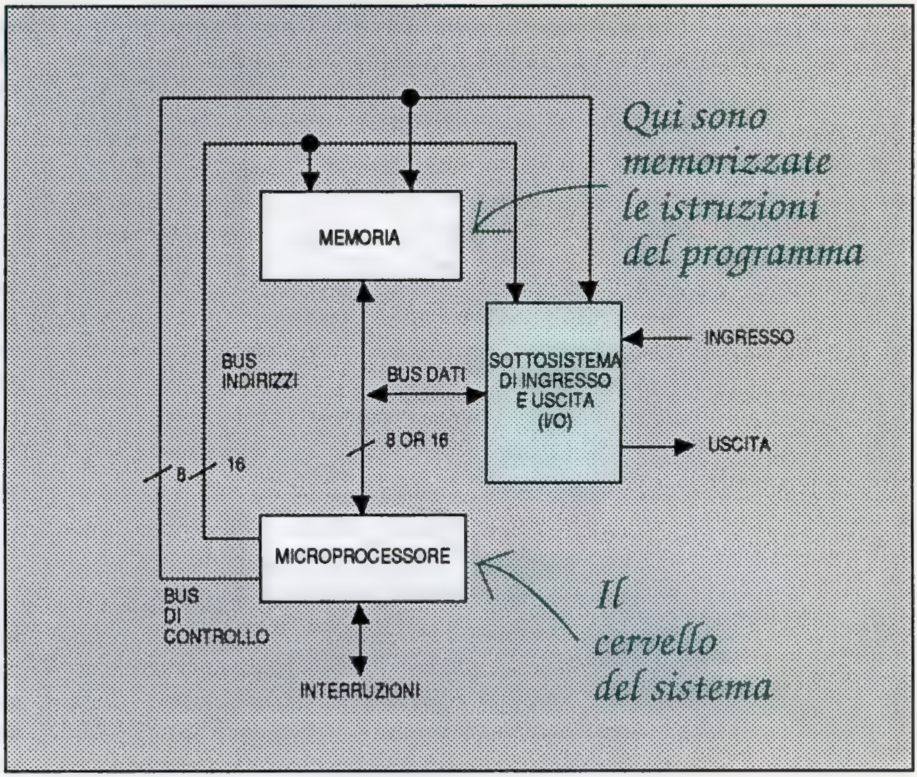
Mentre i componenti di un microcalcolatore possono essere internamente circuiti molto complessi, il loro comportamento funzionale è molto semplice, il che semplifica notevolmente la progettazione del sistema. Comunque, dato che tali dispositivi operano soprattutto su codici di dati digitali in parallelo (molti segnali appaiono su una serie di fili contemporaneamente), il compito del riparatore è complicato dal fatto che è necessario conoscere lo stato di molti segnali contemporaneamente. L'analisi e l'identificazione del guasto vengono anche resi più complicati dal fatto che la sequenza di operazioni che la macchina deve compiere non è fissata dai collegamenti di porte e di decodificatori, ma è memorizzata nel sistema come una serie di istruzioni codificate. Con un sistema di questo tipo il riparatore non solo deve verificare il funzionamento del componente hardware e del cablaggio, ma deve anche capire il funzionamento del programma del computer. In molti casi questi compiti possono richiedere l'uso di una strumentazione di test specifica. Prima di poter affrontare l'argomento dei guasti nei microcomputer devono essere capiti il funzionamento e la struttura fondamentale di un microcomputer. Questo sarà l'oggetto della parte restante del capitolo.

Componenti e funzionamento del microcalcolatore

I sottosistemi fondamentali del microcomputer sono il microprocessore, la memoria e le sezioni di ingresso/uscita.

Uno schema a blocchi dei tre principali sottosistemi di un microcomputer è mostrato in *figura 5.11*. L'elemento centrale di controllo è il microprocessore. Esso determina il funzionamento di tutti gli altri componenti del sistema ed ha anche il compito di fornire la sequenza di comportamento della macchina come determinato dal programma memorizzato nel sistema. La memoria del sistema immagazzina i codici binari che rappresentano il programma del computer e i dati, le costanti ed i messaggi del sistema. Il sottosistema di ingresso e di uscita (solitamente indicato con I/O, ossia input/output) fornisce le comunicazioni e i segnali di controllo da e verso l'ambiente esterno.

Figura 5.11. Struttura di un microcomputer


MEMORIA
Qui sono memorizzate le istruzioni del programma
SOTTOSISTEMA DI INGRESSO E USCITA (I/O)
INGRESSO
USCITA
BUS INDIRIZZI
BUS DATI
8 OR 16
8
16
MICROPROCESSORE
BUS DI CONTROLLO
INTERRUZIONI
Il cervello del sistema

Un tipico microcomputer di controllo è formato da un microprocessore, da una o due memorie e da uno o più circuiti di controllo di ingresso/uscita insieme ai circuiti di interfaccia adeguati per collegare l'attrezzatura e i segnali esterni con i circuiti di ingresso e di uscita del microcomputer. In alternativa la memoria, il sottosistema fondamentale di I/O e il microprocessore possono essere acquistati come circuito integrato singolo chiamato microcomputer. In questo caso il sistema sarebbe formato da questo microcomputer e dai circuiti di interfaccia addizionali che possono essere necessari per collegare il circuito del microcomputer con i dispositivi e i circuti esterni. Un microcomputer più grande, ad esempio un personal computer, comprenderebbe uno o più microprocessori, molti circuiti di memoria, molti circuiti di ingresso e di uscita e circuiti di interfaccia associati, una tastiera e un terminale CRT, una o più unità dischi e una stampante. Comunque tutti questi tipi di microcomputer userebbero ancora la stessa struttura di base mostrata in *figura 5.11* con gli stessi tipi di componenti usati in tutti questi sistemi.

Funzionamento della memoria

La memoria è organizzata in parole di 8 o 16 bit (byte).

La memoria è organizzata come un gran numero di locazioni a 8 bit in cui ogni gruppo di 8 bit viene detto byte. Questi byte vengono trasferiti uno alla volta (o due alla volta a seconda del tipo di processore) da e verso il processore e sotto il controllo di questo. I processori che svolgono il loro lavoro su dati e codici di istruzione a

16 bit internamente vengono chiamati processori a 16 bit. Essi possono trasferire sia uno che due byte alla volta, chiamando parola (word) i codici a due byte.
I processori che svolgono operazioni su codici ad 8 bit internamente vengono chiamati processori ad 8 bit. Essi possono trasferire codici di dati e di istruzioni solo 1 byte alla volta.

Bus dati

I dati vengono trasferiti in entrambe le direzioni tra i sottosistemi lungo un percorso conduttore in parallelo chiamato bus.

L'informazione viene trasferita tra la memoria e il microprocessore e tra i dispositivi di ingresso e uscita e il processore lungo 8 o 16 fili (a seconda del tipo di processore) in parallelo. Questi 8 o 16 fili vengono chiamati bus dati. In *figura 5.11* essi vengono rappresentati con una sola linea avente una freccia su entrambe le estremità per indicare che i codici possono passare sui fili in tutte e due le direzioni. Il numero vicino alla sbarra indica quanti fili vengono rappresentati dalla singola linea.

Scrittura in memoria

I dati vengono trasferiti nella memoria dal processore attraverso una sequenza chiamata operazione di scrittura.

Al fine di trasferire le informazioni dal processore alla memoria, il processore deve compiere un'operazione di scrittura in memoria in cui il dato viene posto sul bus dati dal processore: inoltre quest'ultimo trasmette alla memoria l'indirizzo corretto e i segnali di controllo. (Questo procedimento viene chiamato "accesso alla memoria"). Specificatamente il processore genererà un codice di indirizzo di 16 o più bit (a seconda del processore) per indicare quale locazione della memoria deve ricevere i dati. In seguito, quando il dato è sul bus dati, il processore porterà una linea di abilitazione di memoria allo stato attivo per far compiere alla memoria l'operazione di scrittura, cioè per memorizzare l'informazione.

Lettura dalla memoria

I dati vengono trasferiti fuori dalla memoria nel processore attraverso una sequenza chiamata operazione di lettura.

Il trasferimento di dati in senso contrario, dalla memoria al processore, viene svolto in maniera simile. In questo caso l'operazione è detta lettura dalla memoria e il processore riceverà o il codice per la successiva istruzione del programma del microcomputer che deve essere eseguita o codici di dati provenienti dalla memoria. Il processore disporrà le sue linee dati come linee di ingresso così da poter ricevere i dati dalla memoria e fornirà sulle linee di indirizzo il codice di indirizzo della locazione che deve essere letta. Quindi porterà il segnale di abilitazione della memoria nel suo stato attivo e così pure farà per quello di abilitazione di lettura. Il circuito di memoria disporrà il codice binario richiesto sulle linee dati e il processore memorizzerà questo codice in uno dei suoi registri interni. Se il codice rappresenta l'istruzione successiva che deve essere eseguita dal processore, il codice sarà memorizzato nel registro di istruzioni interno del processore e l'intera operazione di lettura della memoria sarà nota come acquisizione di istruzione o lettura di istruzione (instruction fetch). Se il codice è un codice dati, i dati saranno acquisiti in uno dei registri dati nel processore per ulteriori operazioni e manipolazioni che dipendono dall'istruzione che si sta eseguendo.

Funzionamento dell'ingresso/uscita

Quando il processore sta svolgendo un'operazione di ingresso/uscita indirizza le porte di controllo nelle sezioni di ingresso/uscita proprio come se esse fossero la memoria, abilita l'unità e invia un segnale di controllo per immettere o emettere dati sul bus dati.

I trasferimenti di dati tra i dispositivi di ingresso/uscita e il processore si svolgono in maniera simile a quella con cui avvengono i trasferimenti di memoria. Infatti in molti progetti le posizioni di ingresso e di uscita sono semplicemente considerate come delle ulteriori locazioni di memoria e non c'è differenza tra un trasferimento di ingresso o uscita e un trasferimento di memoria. In questo caso si parla di ingresso/uscita a mappa di memoria (memory mapped). Altri progetti utilizzano un indirizzo e un segnale di abilitazione differenti per dispositivi di ingresso e di uscita. Si dice che tali processori mantengono uno spazio ingresso/uscita separato ed essi vengono detti essere a mappa di ingresso/uscita.

Indipendentemente dal tipo di trasferimento ingresso/uscita utilizzato, il procedimento di trasmissione dei dati dal processore al dispositivo di uscita viene considerato una scrittura di ingresso/uscita ed è molto simile alla scrittura di memoria. Il processore manderà i dati sul bus dati, il codice di indirizzo o posizione per il dispositivo di uscita desiderato sul bus indirizzi e quindi abiliterà le condizioni di scrittura in uscita. Analogamente la lettura di un dispositivo di ingresso consisterà nel processore che dispone le linee del suo bus dati come linee di ingresso, l'indirizzo per il dispositivo di ingresso desiderato sul bus indirizzi e fornisce un segnale di abilitazione per la lettura di ingresso. In risposta a questi segnali i dati verranno posti sul bus dati dal dispositivo di ingresso e il processore memorizzerà quindi i dati in uno dei suoi registri dati interni per completare l'operazione o l'istruzione di trasferimento di ingresso.

Funzionamento del processore

Il processore controlla completamente tutte le operazioni di lettura e scrittura specificando a quale locazione si deve accedere tramite il codice che esso pone sulle linee di indirizzo e abilitando le linee di controllo relative. Perciò se ci sono 16 linee di indirizzo un codice di 16 zeri farà accedere alla locazione 0 (la prima locazione) sia nella memoria che nello spazio ingresso/uscita. Se un codice di tutti 1 fosse posto sulle 16 linee di indirizzo si avrebbe accesso alla 65535esima locazione di memoria. Perciò un codice di indirizzo a 16 bit (solitamente chiamato semplicemente "indirizzo") può localizzare una delle 65536 differenti locazioni o byte in memoria per consentire l'accesso da parte del processore.

Le principali funzioni del processore sono di accedere alla memoria in ogni locazione, di controllare tutte le operazioni di lettura e di scrittura e di svolgere le operazioni che permettono il compiersi di queste funzioni nell'esatta sequenza.

Il processore deve generare l'indirizzo per la locazione sia dell'istruzione successiva che dei dati di cui ha bisogno per eseguire l'istruzione che è presente in quel momento nel suo registro istruzioni. Quindi esso abilita il suo registro di ingresso per un'operazione di lettura o abilita il suo registro di uscita per un'operazione di scrittura. Infine il processore abilita la memoria o il sottosistema ingresso/uscita per svolgere l'operazione di lettura o di scrittura come richiesto.

Questo procedimento viene continuamente ripetuto dal processore mentre esso esegue il programma, un'istruzione dopo l'altra, e intanto che esso accede ai dati necessari per eseguire ciascuna istruzione. Infatti la principale funzione del microprocessore è quella di acquisire ed eseguire istruzioni di un programma nell'ordine esatto così da controllare che la macchina si comporti nel modo desiderato.

UNA VERSIONE MICROCOMPUTERIZZATA DELL'UNITA' DI CONTROLLO DI UN DISTRIBUTORE AUTOMATICO

Un singolo microcomputer contenente una memoria, un processore e circuiti d'ingresso/uscita sostituisce tutti i precedenti circuiti utilizzati per il distributore automatico. Esso deve avere circuiti di interfaccia per accoppiare ingressi ed uscite e un programma da eseguire.

Come esempio del progetto di un microcomputer e delle sue applicazioni ad un problema di controllo reale, considereremo una versione microcomputerizzata del circuito del distributore automatico trattato precedentemente nel capitolo. Questo esempio aiuterà a chiarire alcuni dei concetti introdotti in quest'ultima parte.

L'approccio usato è quello di includere tutti i componenti dello schema a blocchi di *figura 5.11* in un singolo microcomputer a circuito integrato, ad esempio il TMS9940. Questo microcomputer contiene una memoria per i dati e per il programma e inoltre il processore e i circuiti di ingresso/uscita. Il processore è del tipo a 16 bit in quanto elabora i codici di istruzioni e i codici dati interni 16 bit alla volta. Poiché la memoria e i circuiti fondamentali di ingresso e di uscita sono tutti interni al dispositivo, non c'è bisogno di un bus indirizzi o di un bus dati esterno.

Per utilizzare il microcomputer come unità di controllo di un distributore automatico devono essere collegati col TMS9940 solo gli ingressi esterni del distributore automatico. Quindi può essere scritto e memorizzato un programma per generare i segnali di controllo del distributore automatico in modo tale da provocare la sequenza di comportamento mostrata in *figura 5.8*.

Collegamenti hardware

I collegamenti hardware fondamentali sono mostrati in *figura 5.12*. Le uscite dell'identificatore di monete N, D, Q e il segnale risultante $\overline{C}$ vengono mandati al microcomputer. Ciascuno dei pulsanti di selezione dei trenta articoli è collegato al circuito codificatore a priorità di linea 32-a-5. Questo circuito genera un codice binario a 5 bit se viene premuto il pulsante a priorità più alta. Questi 5 bit vengono inseriti nel microcomputer e vengono inviati anche al decodificatore di linea 5-a-32. L'uscita del decodificatore, quando viene abilitata dal segnale $\overline{\text{VEND}}$, provoca l'emissione dell'articolo scelto. Il segnale $\overline{S}$ viene generato quando uno qualsiasi dei pulsanti di selezione è chiuso. Il circuito di accensione e di azzeramento a pulsante è lo stesso della precedente versione.

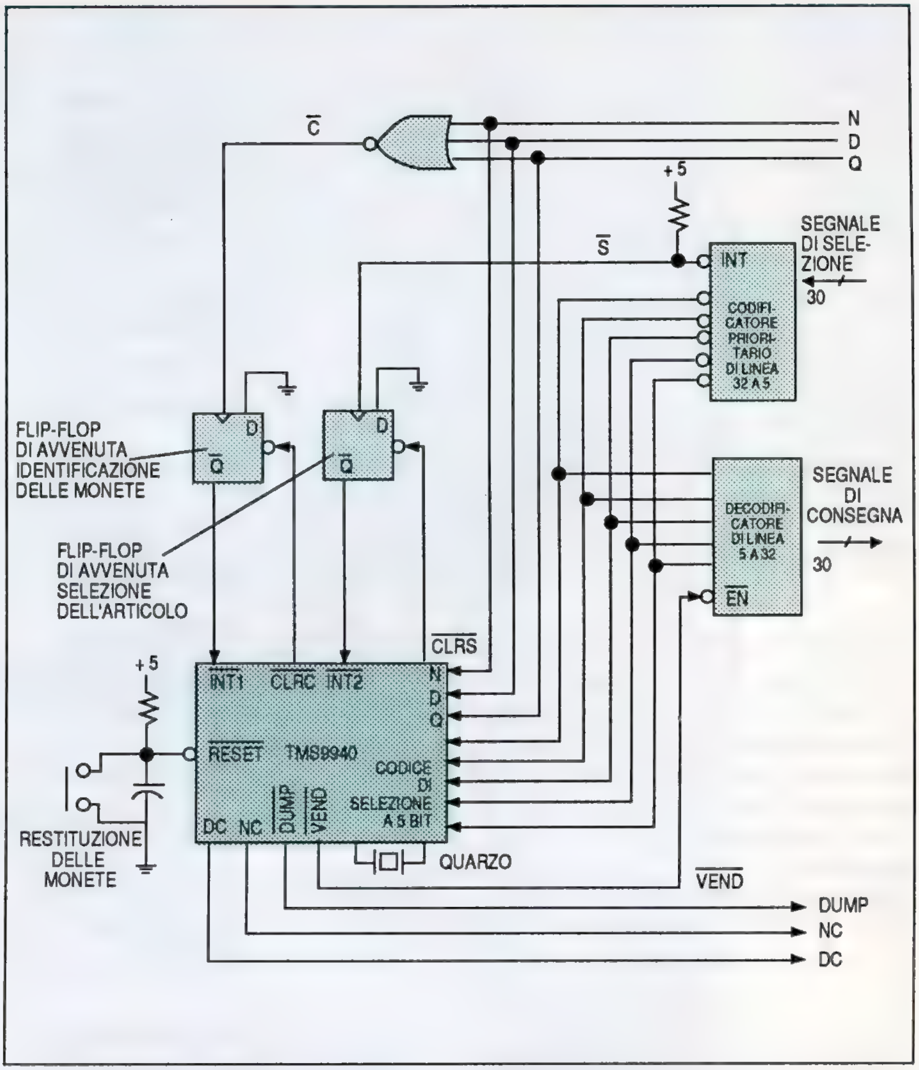
Il microcomputer genera le uscite DUMP per la restituzione delle monete se il totale delle monete inserite è minore del valore dell'articolo, $\overline{\text{VEND}}$ per l'emissione dell'articolo scelto, NC e DC per la restituzione del resto. Il microcomputer genera anche uscite per azzerare i flip-flop che hanno registrato le monete inserite e l'articolo scelto. Questi flip-flop generano segnali detti interruzioni. Il segnale $\overline{\text{INT1}}$ interromperà il microcomputer per notificargli che è stata inserita una moneta. Analogamente il segnale $\overline{\text{INT2}}$ interromperà il microcomputer per fargli sapere che è stato premuto un pulsante di selezione dell'articolo. L'interruzione farà sì che il programma del computer risponda agli eventi $\overline{C}$ o $\overline{S}$ nel modo indicato dalla sequenza di *figura 5.8*.

Programma

Il programma memorizzato nel

Lo schema di *figura 5.12* mostra ingressi e uscite del segnale, ma non indica come il microcomputer risponda ad uno qualsiasi dei segnali di ingresso o sotto quali con-

Figura 5.12. Logica di un distributore automatico con microcomputer


$\overline{C}$
N
D
Q
+5
$\overline{S}$
SEGNALE DI SELEZIONE
INT
CODIFICATORE PRIORITARIO DI LINEA 32 A 5
30
FLIP-FLOP DI AVVENUTA IDENTIFICAZIONE DELLE MONETE
D
$\overline{Q}$
D
$\overline{Q}$
FLIP-FLOP DI AVVENUTA SELEZIONE DELL'ARTICOLO
DECODIFICATORE DI LINEA 5 A 32
$\overline{EN}$
SEGNALE DI CONSEGNA
30
$\overline{CLRS}$
+5
$\overline{INT1}$
$\overline{CLRC}$
$\overline{INT2}$
N
D
Q
$\overline{RESET}$
TMS9940
CODICE DI SELEZIONE A 5 BIT
DC
NC
$\overline{DUMP}$
$\overline{VEND}$
RESTITUZIONE DELLE MONETE
QUARZO
$\overline{VEND}$
DUMP
NC
DC

dizioni genera uno qualsiasi dei segnali di uscita. Questa informazione è contenuta nel programma del computer che è memorizzato nella memoria interna al microcomputer. Solo esaminando il programma e mettendolo in relazione col funzionamento della macchina si può verificare che lo schema di *figura 5.12* si comporterà come quelli delle *figure 5.3, 5.4, 5.5, 5.6, 5.7* e *5.8*. Tutto quello che si può determinare dalla configurazione hardware di figura 5.12 è che si tratta di una configurazione molto semplice e che i segnali aspettati vengono controllati e generati.

Se fosse esaminato il programma contenuto nella memoria del microcomputer si vedrebbe che esso è composto da molti byte di codici di istruzione formati da 0 ed 1 binari detti codici macchina. Questi codici devono essere convertiti in una forma detta linguaggio assemblativo (assembly language) in modo che possano essere com-

microcomputer dirige il microcomputer stesso nel controllare gli ingressi, nell'eseguire le istruzioni sulla base degli ingressi ricevuti e nel generare i segnali d'uscita per svolgere l'azione richiesta.

Figura 5.13. Programma di un distributore automatico

Spiegazione

Abbreviazione dell'operazione svolta

```
Etichetta   Mnemonico   Operando    Commento
  S0          CLR         R1          Azzera il registro TOTAL
              CLR         R2          Azzera il registro VALUE
              LI          12, >3C0    Predispone l'indirizzo di base di I/O
              SBZ         16          Azzera il flip-flop SELECT
              SBO         16
  S3          SBZ         17          Azzera il flip-flop COIN
              SBO         17
              LIIM        3           Attiva gli interrupt S (S̄) e C (C̄)
  S1          IDLE                    Aspetta un segnale di interrupt
  S2          TB          8           Controlla l'ingresso di nickel
              JEQ         NL
              TB          9           Prova l'ingresso di dime
              JEQ         DI
              TB          10          Prova l'ingresso di quarter
              JNE         S3
              JMP         QT
  NL          AI          1, 5        Somma 5 cent a TOTAL
              DCA
              JMP         S3          Vai allo stato 3
  DI          AI          1, >10      Somma 10 cent a TOTAL
              DCA
              JMP         S3          Vai allo stato 3
  QT          AI          1, >25      Somma 25 cent a TOTAL
              DCA
              JMP         S3          Vai allo stato 3
  S4          STCR        3, 5        Ottieni il codice SELECT
              MOV         VAL(3), 2   Ottieni VALUE dell'articolo dalla tabella in R2
              C           2, 1        Confronta VALUE dell'articolo con TOTAL
              JLE         VD          Va a VEND se VALUE =< TOTAL
              SBO         18          Genera DUMP
              SBZ         18
              JMP         S0          Va allo stato 0 (S0) se VALUE > TOTAL
  VD          SBZ         19          Genera VEND
              SBO         19
              JEQ         S0          Se VALUE = TOTAL vai allo stato 0 (S0)
  S5          S           2, 1        Altrimenti sottrai VALUE da TOTAL
              LI          4, 1
              COC         4, 1        Se la differenza è dispari
              JEQ         NC          Vai a NC per emettere nickel
  DC          SBO         21          Altrimenti emetti dime
              SBZ         21          Generando DC
  D?          LI          3, >10      Sottrai 10 cent dalla differenza
              S           3, 1
              DCS
              JEQ         S0          SE 0, vai allo stato 0 (S0)
```

Figura 5.13. Programma di un distributore automatico (continua)

```
S6      SBO     21        Altrimenti emetti un altro dime
        SBZ     21
        JMP     S0        Vai allo stato 0 (S0)
NC      SBO     20        Emetti nickel
        SBZ     20        generando NC
        LI      3, >5     Sottrai 5 cent dalla differenza
        S       3, 1
        DCS
        CI      1, 0
        JNE     56
        JMP     50        Verifica la necessità di dime nel resto
```

presi facilmente da una persona che abbia dimestichezza con la serie di istruzioni di questo microcomputer. La conversione di un programma dal codice macchina al linguaggio assemblativo è detta disassemblaggio. Un programma in linguaggio assembly viene scritto come una sequenza di comandi che consistono di abbreviazioni di istruzioni, dette mnemoniche, e di indicatori di posizione dei dati, detti operandi. Un esempio di programma in linguaggio assembly che può essere utilizzato per l'esempio del distributore automatico è mostrato in *figura 5.13* insieme ai relativi commenti esplicativi. Il lettore dovrebbe esaminare questo programma dettagliatamente per capire che esso racchiude il comportamento dello schema originale descritto in *figura 5.8.*

Linguaggio assembly

In un programma in linguaggio assemblativo ciascuno dei quattro campi di un comando nel linguaggio fornisce un certo tipo di informazioni circa l'istruzione.

Al fine di comprendere il programma di un microcomputer scritto sotto forma di linguaggio assemblativo è necessario capire le varie parti dei comandi di questo linguaggio. Un comando in linguaggio assemblativo ha la seguente forma generale:

Etichetta Mnemonico Operando Commento

I campi Etichetta e Commento sono facoltativi. Il campo Mnemonico è sempre richiesto e il campo Operando può essere o no necessario a seconda dell'istruzione. La prima istruzione di *figura 5.13* presenta tutti e quattro i campi.

L'Etichetta è un nome della posizione di una istruzione nella memoria del microcomputer e viene utilizzata per ogni istruzione alla quale si può accedere partendo da un'altra istruzione. Essa può anche essere usata per indicare l'inizio di una parte importante del programma. Per esempio in *figura 5.13* l'Etichetta della prima istruzione è S0 che corrisponde allo stato 0 di *figura 5.8*. Perciò la sequenza di istruzioni che

Il mnemonico è un'abbreviazione della funzione dell'istruzione che viene utilizzata per aiutare la memoria del programmatore. L'operando indica su che cosa si sta lavorando. Il commento è una spiegazione in un linguaggio abbreviato.

comincia da quella etichettata con S0 e finisce con quella appena prima dell'istruzione etichettata con S1 soddisferà i requisiti dello stato S000 di *figura 5.8*.
Il Mnemonico è un'abbreviazione dell' operazione che viene eseguita. Nella prima istruzione del programma di *figura 5.13* il mnemonico è CLR, che è un'abbreviazione per l'operazione:

> Azzera (CLeaR) i contenuti della locazione di memoria specificata nel campo operando.

Il campo Operando indica quale locazione di memoria è implicata nell'istruzione. Nella prima istruzione di *figura 5.13* è la locazione nota come R1. Ciò significa che si tratta del registro 1 della serie dei registri di lavoro del processore, che ne comprende 16.
Il Commento è semplicemente la spiegazione verbale di ciò che sta facendo l'istruzione. Nella prima istruzione di *figura 5.13* il commento è:

> Azzera il registro TOTAL

In altre parole al registro 1 è stato assegnato il compito di tenere il totale delle monete inserite ed esso viene quindi azzerato dalla prima istruzione del programma. La seconda istruzione di questa sequenza utilizza lo stesso mnemonico per azzerare il registro 2 (R2) al quale è stato assegnato il compito di registrare il valore dell'articolo scelto. Perciò i commenti di un programma sono la descrizione di ciò che il programma sta facendo.
Un programma del tipo mostrato in *figura 5.13* può essere compreso grazie ai commenti anche da un lettore che non abbia alcuna familiarità con il significato delle abbreviazioni delle istruzioni mnemoniche e degli operandi utilizzate nel codificare il linguaggio assembly. Per esempio, il lettore potrebbe non conoscere il mnemonico SBZ o il mnemonico SBO delle istruzioni quarta e quinta di *figura 5.13*, ma il commento indica che queste istruzioni vengono utilizzate per azzerare il flip-flop di selezione. Immediatamente dopo questi mnemonici, nella sesta e nella settima istruzione, si ha un'altra coppia di SBZ e SBO che, come mostra il commento, viene utilizzata per azzerare il flip-flop delle monete. A questa coppia viene assegnata l'etichetta S3. Si noti che il programma salta ad S3 altre volte per azzerare il flip-flop delle monete attraverso le istruzioni di salto JNE e JMP.
Lo scopo dell'istruzione SBZ è quello di portare a 0 il bit indicato (il bit 16 per il flip-flop di selezione e il bit 17 per il flip-flop delle monete) mentre lo scopo di SB0 è quello di portare il bit indicato ad 1. Questo fa sì che venga inviato uno 0 seguito da un 1 all'ingresso $\overline{CLR}$ del flip-flop. In altre parole viene inviato un impulso basso momentaneo al terminale $\overline{CLR}$ del flip-flop per azzerare quest'ultimo; quindi rimane un livello alto cosicché il flip-flop possa funzionare.
Le seguenti spiegazioni dei mnemonici utilizzati in *figura 5.13* dovrebbero aiutare il lettore a comprendere il funzionamento fondamentale di ogni sequenza di stato compresa fra S0 ed S6. Queste vengono elencate in ordine alfabetico.

AI	R,	valore Somma il valore ai contenuti del registro R.
C	R1, R2	Confronta i contenuti del registro R1 con quelli del registro R2.
CLR		Azzera la locazione di memoria nel campo operando portandola a tutti 0.
DCA		Svolgi la somma precedente in modo corretto per i codici decimali (BCD).
IDLE		Aspetta un segnale di interruzione.
JEQ	Etichetta	Come JNE solo che il salto viene effettuato se l'ultimo risultato era uguale a 0. Se l'ultima operazione era un'operazione TB, si avrà il salto solo se il bit controllato era ad 1.
JMP	Etichetta	Passa ad eseguire l'istruzione successiva all'etichetta
JNE	Etichetta	Se l'ultimo risultato non era uguale a 0 (a meno che non si trattasse di un'operazione TB), passa ad eseguire l'istruzione successiva all'etichetta; altrimenti esegui l'istruzione successiva all'istruzione JNE.
LI	R,	valore Metti il valore nel registro R.
LIIM	n	Imposta la maschera di interruzione ad n; ciò determina a quali segnali di interruzione si risponderà (livelli n o meno).
MOV	01, 02	Porta i contenuti della locazione 01 nella locazione 02.
S	R1, R2	Sottrai i contenuti del registro R1 dai contenuti del registro R2 e memorizza i risultati nel registro R2.
SBO	n	Porta il bit n ad 1.
SBZ	n	Porta il bit n a 0.
STCR	R, n	Inserisci n bit nel registro R.
TB	n	Controlla il valore del bit numero n; metti il valore nel flip-flop EQ del microcomputer.

CHE COSA ABBIAMO IMPARATO?

1. Il riparatore deve completamente conoscere la sequenza delle operazioni di un sistema logico sequenziale per individuare la causa di un suo comportamento anomalo.
2. Il riparatore deve conoscere dettagliatamente il funzionamento di tutti i circuiti di ingresso e di uscita così come la sequenza delle operazioni che compie l'unità di controllo in risposta ai segnali di ingresso e alle operazioni già svolte.
3. Questa sequenza funzionale viene normalmente descritta con un diagramma di flusso o con un listato di programma passo a passo.
4. Le deviazioni rispetto a questa normale sequenza descritta possono essere messe in relazione con dei problemi nei flip-flop di stato o nei circuiti dei segnali di ingresso.
5. Una volta circoscritto il guasto ad una particolare serie di circuiti di ingresso o di flip-flop di stato, è possibile individuare il guasto utilizzando le tecniche descritte nei capitoli precedenti: cioè attraverso le misurazioni di tensione, resistenza o corrente, attraverso le tecniche di rilevazione del segnale o attraverso le tecniche di verifica del funzionamento di un componente.
6. Se un microcomputer viene utilizzato come unità di controllo, oltre che l'hardware deve essere controllato anche il programma per determinare se esistono alcune imperfezioni all'interno del programma stesso.

Quiz del Capitolo 5

1. Un sistema logico sequenziale ha uscite che dipendono da:
   - a. Solo dagli ingressi al sistema in quel determinato momento.
   - b. Dagli ingressi attuali e da quelli passati.
   - c. Dalle uscite del dispositivo di memoria indipendentemente dagli ingressi del sistema.

2. Nella logica di selezione dell'articolo del distributore automatico (*figura 5.3*), viene ricevuto un impulso sulle linee S1 ed S30. In che condizione sarà $\overline{V30}$?
   - a. Basso
   - b. Alto

3. Quando si riceve il segnale VEND nell'esempio del problema 2, che articolo verrà emesso?
   - a. L'articolo 1
   - b. L'articolo 2
   - c. L'articolo 30

4. Nell'esempio del problema 3, quale sarà il valore V_4 di *figura 5.4*?
   - a. Basso
   - b. Alto

5. Se un acquirente inserisce due nickel e un quarto di dollaro, quale sarà l'uscita del registro a 8 bit all'istante del segnale $\overline{S}$ di *figura 5.5*?
   - a. 35
   - b. 00100011
   - c. 00110101
   - d. 01010011

6. Nell'unità di controllo di *figura 5.6* quale stato causerà un'uscita bassa sulla linea $\overline{CLR}$?
   - a. Lo stato 0
   - b. Lo stato 3
   - c. Lo stato 7
   - d. Lo stato 2

7. Nel circuito di *figura 5.6*, quale stato provocherà un segnale VEND?
   - a. Lo stato 4
   - b. Lo stato 5
   - c. Lo stato 6
   - d. Lo stato 7

8. Nel circuito di *figura 5.6*, se lo stato attuale è lo stato 3, sotto quali condizioni di ingresso lo stato successivo sarà lo stato 4?
   - a. T>V è alto
   - b. $\overline{CRRLOIN}$ è basso
   - c. S è alto
   - d. In nessun caso

9. Sotto quali condizioni la macchina descritta in *figura 5.8* potrebbe passare dallo stato 5 allo stato 7?
   - a. Sotto nessuna condizione.
   - b. Se passando dallo stato 5 allo stato 6 il secondo flip-flop cambia più velocemente del terzo flip-flop.
   - c. Se passando dallo stato 5 allo stato 6 il secondo flip-flop cambia più lentamente del terzo flip-flop.
   - d. Se passando dallo stato 5 allo stato 0 il primo flip-flop cambia più velocemente del terzo flip-flop.

10. Se nel circuito di clock di *figura 5.9a* il condensatore fosse portato a 0,5 microfarad, quale frequenza si avrebbe?
    - a. 10 hetrz
    - b. 20 hertz
    - c. 200 hertz
    - d. 1000 hertz

11. Lo scopo del bus di controllo di *figura 5.10* è quello di:
    - a. Determinare la locazione nel sistema a cui si deve accedere.
    - b. Fornire al processore comunicazioni provenienti dal mondo esterno.
    - c. Determinare quando devono avvenire letture o scritture in memoria o in ingresso/uscita.
    - d. Fornire al processore codici dati e codici istruzioni.

12. Nel programma di *figura 5.12* l'istruzione AI rappresenta il componente hardware:
    - a. Del comparatore.
    - b. Della logica di selezione dell'articolo.
    - c. Dell'addizionatore.
    - d. Della logica di identificazione delle monete.

Aspetti delle memorie

SCOPO DEL CAPITOLO

I principi fondamentali del funzionamento del microcomputer e dei suoi sottosistemi sono stati introdotti nell'ultimo capitolo. Un sistema a microcomputer, come una speciale unità di controllo o un personal computer, può avere una memoria piuttosto estesa che comprende dispositivi di memoria a sola lettura e dispositivi di memoria ad accesso casuale di cui si è già parlato nel capitolo 2. In questo capitolo verranno trattati i principi di funzionamento di questi sottosistemi di memoria e le tecniche speciali per trovare un guasto in essi: è necessario avere queste conoscenze per poter ricercare i guasti nei moderni sistemi a microcomputer.

MEMORIA DI LETTURA-SCRITTURA

La memoria di lettura e scrittura è solitamente un dispositivo RAM. Supponendo che non venga mai staccata l'alimentazione, le RAM statiche mantengono indefinitamente i dati che sono stati immessi in esse, ma le RAM dinamiche devono essere ripristinate ogni 1 o 2 millisecondi.

I dispositivi di memoria di lettura-scrittura, solitamente chiamati RAM (Random access memory), sono formati da una matrice di celle di immagazzinamento. Un bit può essere scritto in una cella o può essere letto da questa. Per conservare i bit memorizzati nelle celle si dovrà alimentare la RAM in maniera continua.

Le celle possono essere composte da semplici flip-flop R-S (trattati nel capitolo 1) o da speciali celle di immagazzinamento di carica formate da transistori di abilitazione e condensatori. In entrambi i casi la matrice è una struttura rettangolare che ha un certo numero di righe ognuna delle quali contiene un certo numero di colonne di celle di memoria. Le RAM che utilizzano celle di memoria a flip-flop vengono chiamate RAM statiche, dato che esse mantengono i bit memorizzati senza la necessità di ulteriori azioni dopo che il flip-flop è stato impostato. Le RAM che utilizzano invece il meccanismo di memorizzazione di carica vengono dette RAM dinamiche (DRAM). Queste ultime devono essere periodicamente ripristinate, solitamente una volta ogni uno o due millisecondi, per conservare l'informazione memorizzata. Questa operazione di ripristino è necessaria per ricaricare il condensatore, poiché questo altrimenti si scaricherebbe gradualmente. Se non si ripristinasse l'informazione memorizzata con la frequenza necessaria si provocherebbe la perdita dell'informazione stessa. La necessità o meno di ripristinare le informazioni è la differenza principale tra la memoria dinamica e la memoria statica.

Memorie statiche

Cella singola

Una tipica cella di memoria statica è mostrata in *figura 6.1*. La struttura della cella per questo tipo di memoria è il flip-flop formato da una coppia di porte NAND incrociate $\overline{GQ}$ e GQ. I dati memorizzati nella cella verranno emessi sulla linea dati se si sta compiendo un'operazione di lettura (con $\overline{S}$ e $\overline{G}$ contemporaneamente bassi men-

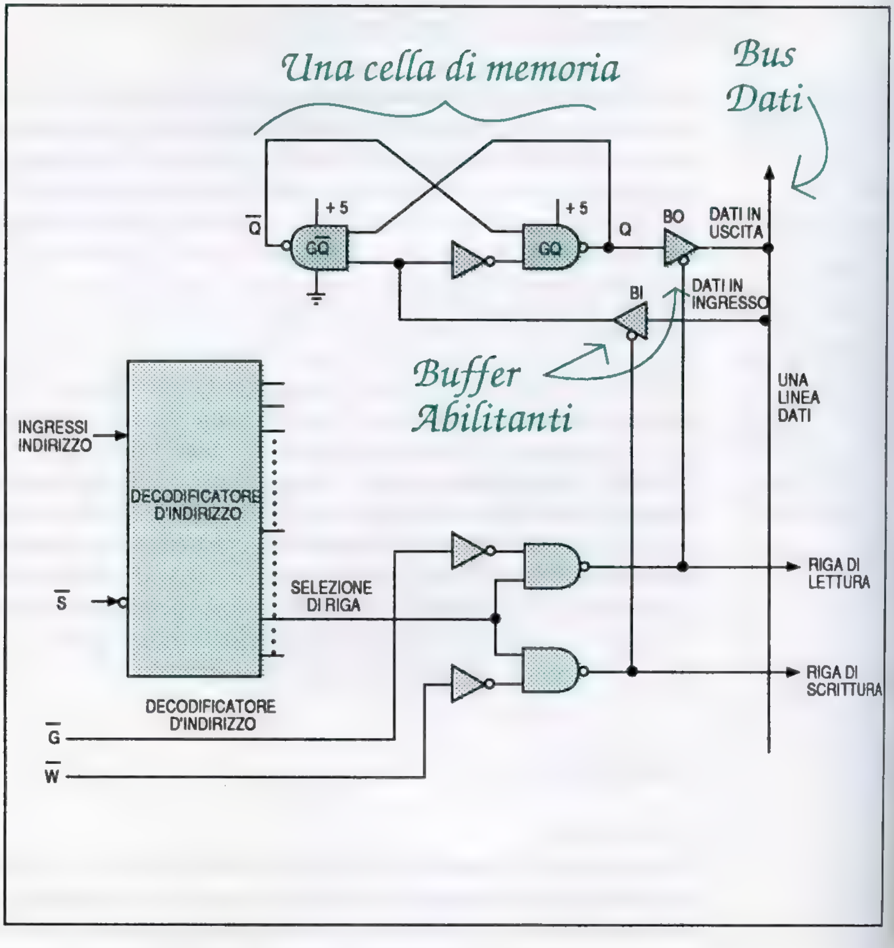
Per un'operazione di lettura il decodificatore d'indirizzo abilita il buffer a tre stati a passare i bit della cella di memoria al bus. Per un'operazione di scrittura il decodificatore d'indirizzo abilita il buffer a tre stati a passare i bit sul bus alla cella di memoria.

tre $\overline{W}$ è alto). Analogamente i dati sulla linea dati saranno memorizzati nella cella se si sta compiendo un'operazione di scrittura (con $\overline{S}$ e $\overline{W}$ contemporaneamente bassi mentre $\overline{G}$ è alto).

Il passaggio dei dati da e verso la linea dati viene controllato da buffer a tre stati (BO e BI), i quali a loro volta sono controllati da porte (rispettivamente GR e GW) che abilitano i buffer quando si verificano le corrette condizioni $\overline{S}$, $\overline{G}$ e $\overline{W}$. Se si sta compiendo un'operazione di lettura e la cella mostrata si trova nella riga selezionata dal decodificatore di indirizzo con $\overline{S}$ basso, l'uscita della porta di controllo GR sarà bassa. Ciò farà sì che BO venga portato nello stato per cui passi il valore (0 od 1) di Q sulla linea dati. Le uscite delle altre sette celle di memorizzazione della stessa riga (che non sono mostrate) saranno collegate con le altre sette linee dati per ottenere sul bus dati un codice a 8 bit completo.

Se invece si sta compiendo un'operazione di scrittura e la cella mostrata si trova nella riga selezionata con $\overline{S}$ basso, l'uscita della porta di controllo GW sarà bassa. Questo metterà BI in grado di passare il dato sulla linea dati all'ingresso della cella di me-

Figura 6.1. Cella di memoria statica


Una cella di memoria
Bus Dati
$\overline{Q}$
+5
$G\overline{Q}$
+5
GQ
Q
BO
DATI IN USCITA
BI
DATI IN INGRESSO
Buffer Abilitanti
UNA LINEA DATI
INGRESSI INDIRIZZO
DECODIFICATORE D'INDIRIZZO
$\overline{S}$
SELEZIONE DI RIGA
RIGA DI LETTURA
RIGA DI SCRITTURA
DECODIFICATORE D'INDIRIZZO
$\overline{G}$
$\overline{W}$

moria (essenzialmente agli ingressi R ed S del flip-flop). Se questa condizione perdura abbastanza a lungo cosicché il flip-flop possa rispondere, lo 0 o l'1 sulla linea dati verrà memorizzato nella cella. Nello stesso tempo i dati sulle altre sette linee dati verranno memorizzati nelle altre sette celle della linea scelta.

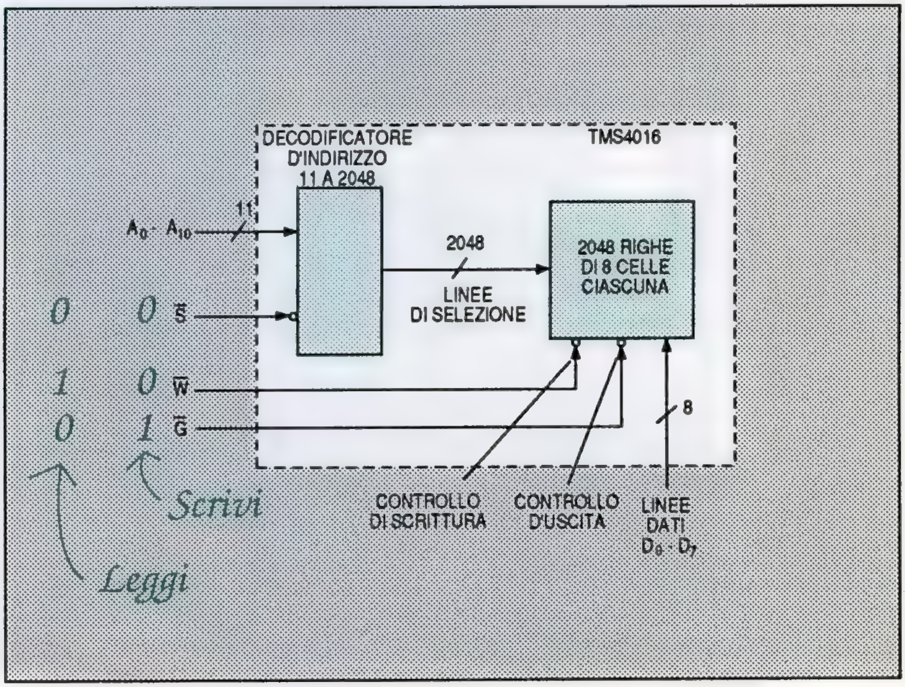
RAM a circuito integrato

Sia per scrivere che per leggere da una memoria di 2048 parole, un codice d'indirizzo ad 11 bit localizza la parola di memoria desiderata tramite un decodificatore d'indirizzo abilitato. Per leggere dalla memoria il segnale $\overline{G}$ deve essere mantenuto basso; per scrivere deve essere invece

La *figura 6.2* mostra una RAM statica a circuito integrato TMS4016 sotto forma di schema a blocchi. Questo circuito possiede 16.834 celle simili a quelle di *figura 6.1*. Le celle sono organizzate in 2048 righe con 8 celle per riga cosicché ciascuna riga di celle memorizzi un byte di informazione. Allo scopo di determinare da quale linea di memoria si stia leggendo o su quale si stia scrivendo, il sistema deve fornire al circuito un codice di indirizzo a 11 bit ($2^{11} = 2048$). I segnali $\overline{G}$ e $\overline{W}$ indicano al circuito se si sta svolgendo un'operazione di lettura o di scrittura; comunque non si possono verificare operazioni di lettura o scrittura se non è applicato uno 0 logico all'ingresso di controllo $\overline{S}$ (Select-not).

Per leggere l'informazione da una data riga, l'indirizzo a 11 bit deve essere messo sulle linee dalla A_0 alla A_{10}, e le linee $\overline{S}$ e $\overline{G}$ devono essere portate ad un livello basso (0 logico) mentre $\overline{W}$ viene mantenuto ad un livello alto (1 logico). Il byte ad 8 bit verrà inviato sulle linee da D_0 a D_7 (il bus dati) nell'intervallo di tempo in cui sia $\overline{S}$ che $\overline{G}$ vengono mantenuti bassi. In questo periodo di tempo il codice di indirizzo sulle linee tra A_0 ed A_{10} deve essere mantenuto stabile e nessun altro dipositivo del sistema deve tentare di inviare degli 1 o degli 0 sul bus dati.

Figura 6.2. RAM statica TMS4016



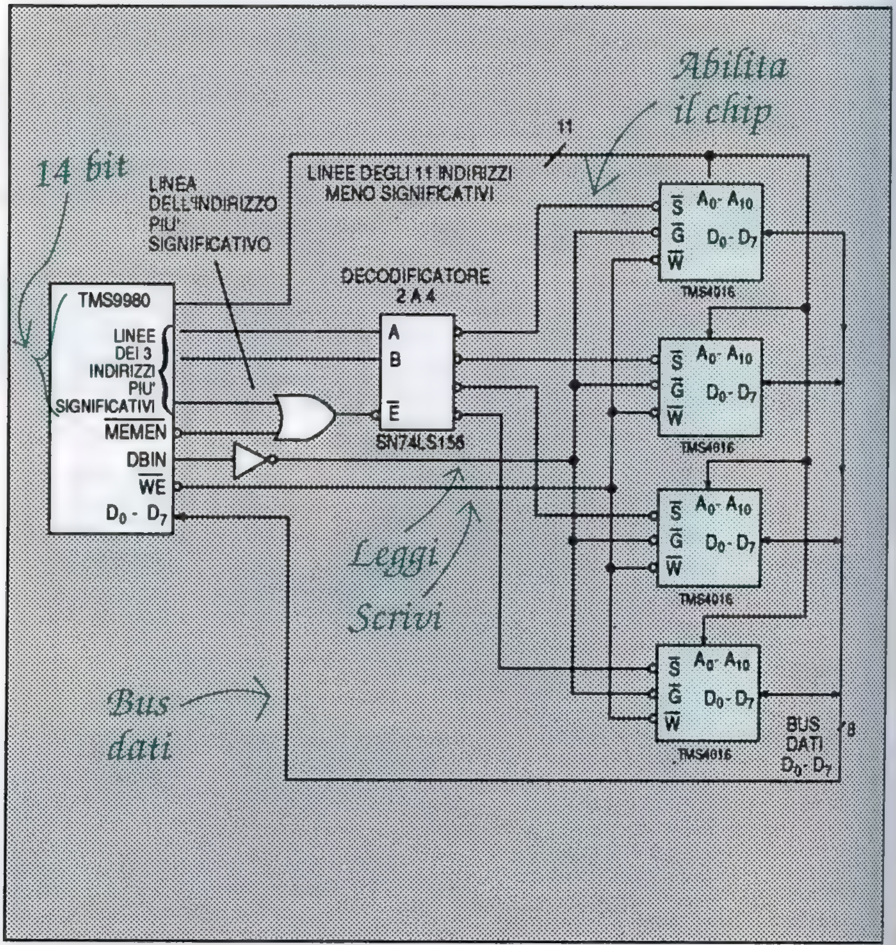
tenuto basso il segnale $\overline{W}$.

Una sequenza di segnali simile a quella appena descritta viene richiesta per scrivere nella memoria. Il codice di indirizzo ad 11 bit viene posto sulle linee A_0 e A_{10} e il dato che deve essere memorizzato viene posto sulle linee dalla D_0 alla D_7 dal processore del sistema o da altri dispositivi di controllo della memoria. Quindi le linee $\overline{S}$ e $\overline{W}$ vengono portate a valori bassi mentre $\overline{G}$ viene mantenuto alto per un tempo sufficientemente lungo da permettere che l'informazione venga memorizzata nella riga di celle selezionata. Una volta trascorso questo tempo i segnali S e W ritornano ai loro livelli alti inattivi, i dati possono essere rimossi dalle linee da D_0 a D_7 e il codice di indirizzo sulle linee dalla A_0 alla A_{10} può essere cambiato.

Sottosistema di memoria

Il circuito mostrato in *figura 6.3* è una memoria di 8192 byte formata da 4 dispositivi di memoria a circuito integrato da 2048 byte del tipo riportato in *figura 6.2*. I circuiti esterni devono fornire i segnali compresi tra A_0 e A_{10}, $\overline{S}$, $\overline{W}$ e $\overline{G}$. Inoltre devo-

Figura 6.3. Sottosistema di memoria statica composto da TMS4016


Abilita il chip
14 bit
11
LINEA DELL'INDIRIZZO PIU' SIGNIFICATIVO
LINEE DEGLI 11 INDIRIZZI MENO SIGNIFICATIVI
TMS9980
LINEE DEI 3 INDIRIZZI PIU' SIGNIFICATIVI
$\overline{MEMEN}$
DBIN
$\overline{WE}$
D_0 - D_7
DECODIFICATORE 2 A 4
A
B
$\overline{E}$
SN74LS156
$\overline{S}$
$\overline{G}$
$\overline{W}$
A_0 - A_{10}
D_0 - D_7
TMS4016
Leggi
Scrivi
Bus dati
BUS DATI D_0 - D_7
8

no fornire i dati sulle linee comprese tra D_0 e D_7 durante un'operazione di scrittura o ricevere i dati dalla memoria sulle linee comprese tra D_0 e D_7 durante un'operazione di lettura. Il microprocessore di solito fornisce questi segnali esterni.

Il microprocessore TMS9980 di *figura 6.3* fornisce tipicamente 14 bit di codice di indirizzo. Gli 11 bit meno significativi di questi indirizzi vengono collegati con gli ingressi di indirizzo compresi fra A_0 ed A_{10}. Le 3 linee di indirizzo più significative ed il segnale di abilitazione di memoria ($\overline{\text{MEMEN}}$) sono collegati con il decodificatore di indirizzo 2 a 4 per generare il segnale $\overline{\text{S}}$ al fine di abilitare uno specifico circuito integrato di memoria. Il processore genera anche segnali di controllo di lettura e scrittura per fornire i segnali $\overline{\text{G}}$ e $\overline{\text{W}}$ per il dispositivo di memoria.

Le linee dei circuiti di memoria comprese tra D_0 e D_7 e il processore sono collegati tra loro, cosicché i dati possono passare dall'uno all'altro. La linea $\overline{\text{WE}}$ (Write Enable Not) del processore è collegata con le linee $\overline{\text{W}}$ dei circuiti di memoria e il segnale DBIN (Data Bus In) è collegato attraverso un invertitore con gli ingressi G- dei circuiti di memoria. Nel processore mostrato DBIN è alto per un'operazione di lettura e $\overline{\text{WE}}$ è basso per un'operazione di scrittura. $\overline{\text{MEMEN}}$ è invece basso per entrambe le operazioni di memoria. Quando l'ingresso $\overline{\text{E}}$ del decodificatore è basso una delle sue linee di uscita del decodificatore è bassa, in quanto selezionata dalle linee di indirizzo di ordine alto: perciò uno (e solo uno) degli ingressi $\overline{\text{S}}$ sarà ad un livello basso attivo per abilitare un particolare circuito di memoria.

I tre bit di indirizzo più significativi e il segnale $\overline{\text{MEMEN}}$ abilitano il decodificatore il quale attiva uno dei quattro integrati di memoria. Gli 11 bit d'indirizzo meno significativi selezionano di fatto la particolare parola di memoria in ciascun integrato.

In questo esempio l'indirizzo a 14 bit può selezionare un byte qualsiasi dei 16.384 byte che compongono la capacità totale della memoria. Poiché la memoria mostrata possiede solo 8192 byte il bit di indirizzo più significativo deve determinare se è attivata o meno la memoria. Dato che $\overline{\text{MEMEN}}$ e il bit di indirizzo più significativo vengono inviati attraverso una porta OR al terminale $\overline{\text{E}}$ entrambi devono essere bassi per abilitare il decodificatore e, quindi, abilitare il circuito di memoria scelto. Perciò il codice di indirizzo binario per la memoria da 8192 byte deve sempre cominciare con uno 0 e una memoria da 8192 byte risponderà ad indirizzi binari compresi fra 0000 0000 0000 0000 (0000 nel sistema esadecimale) e 0001 1111 1111 1111 (1FFF nel sistema esadecimale). Se il bit di indirizzo più significativo fosse inviato attraverso un invertitore prima di essere trasmesso alla porta OR, la memoria verrebbe attivata solo se fosse 1. In quel caso la memoria risponderebbe ad indirizzi binari compresi tra 0010 0000 0000 0000 (2000 nel sistema esadecimale) e 0011 1111 1111 1111 (3FFF nel sistema esadecimale). Perciò è solitamente facile esaminare gli ingressi al decodificatore per determinare la gamma dei valori di indirizzo a cui la memoria risponde.

Memorie dinamiche

Le RAM dinamiche servono allo stesso scopo delle RAM statiche; per esempio servono per fornire la memorizzazione e il recupero dei programmi e dei dati del sistema. Per funzionare correttamente la configurazione di una RAM dinamica deve includere i dispositivi per essere periodicamente ripristinata così come le interconnessioni adeguate alle linee dati, alle linee di indirizzo e alle linee di controllo del sistema. La necessità dell'operazione di ripristino è una conseguenza del metodo utilizzato per fornire la memorizzazione binaria nelle memorie dinamiche.

Cella singola

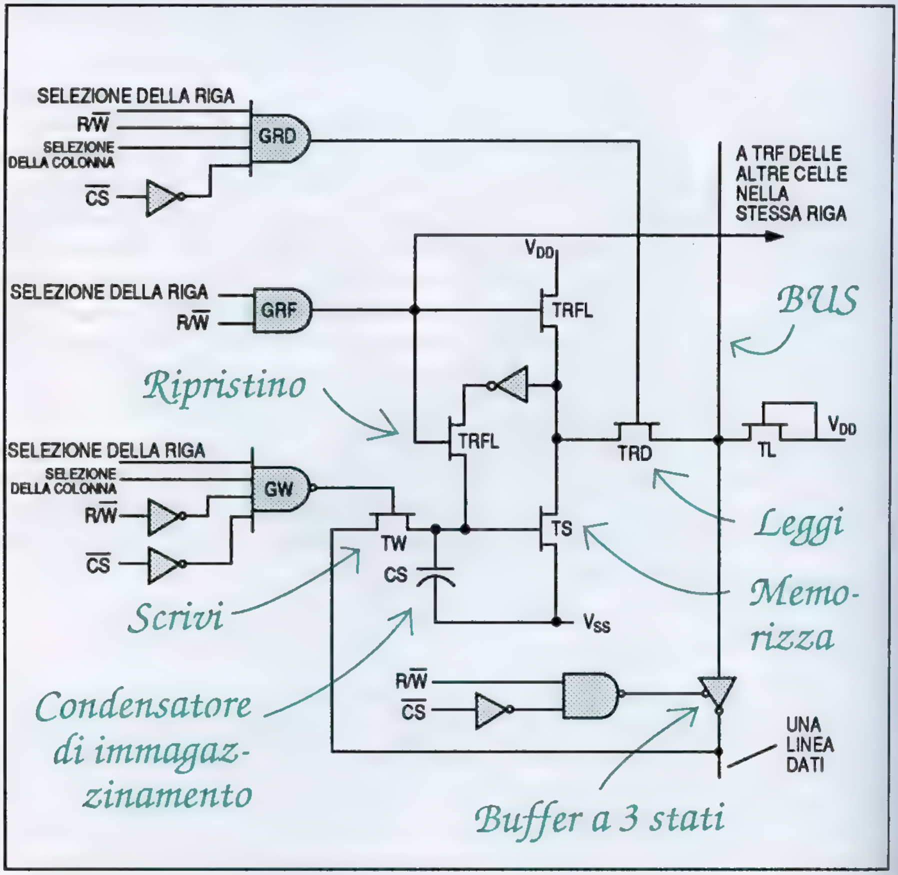
Un condensatore MOS, CS, costituisce la parte centrale di una cella della RAM dinamica. Esso controlla la porta di un transistore di memoria MOS, TS.

La cella di memoria fondamentale è mostrata in *figura 6.4*. Il componente di memoria è il condensatore CS che è parte del transistore di tipo MOS TS. Gli altri transistori passano l'informazione da e verso la cella per operazioni di scrittura, di lettura e di ripristino. Questi transistori sono indicati in *figura 6.4* rispettivamente come TW, TRD e TRF.

Inoltre vi è un sistema di porte simile a quello utilizzato nella RAM statica per impiegare i segnali di selezione della riga, di selezione della colonna e quelli di lettura/scrittura che arrivano alla memoria dai decodificatori per determinare quale cella di immagazzinamento della matrice debba essere attivata. Se i decodificatori selezionano la riga e la colonna della cella di *figura 6.4*, allora le porte NAND di lettura, scrittura e ripristino, rispettivamente GRD, GW e GRF, produrranno l'azione voluta sulla cella di memoria scelta. In tutti i casi sia le linee di selezione della riga che quelle della colonna di tutte le porte NAND devono essere al loro livello attivo (nell'esempio di *figura 6.4* livello alto), perché si verifichi una qualsiasi azione.

Se R/$\overline{W}$ è basso l'uscita della porta GW è bassa per portare in conduzione il transistore TW che collega il condensatore di memoria con la linea di ingresso dati. Que-

Figura 6.4. Cella di memoria dinamica

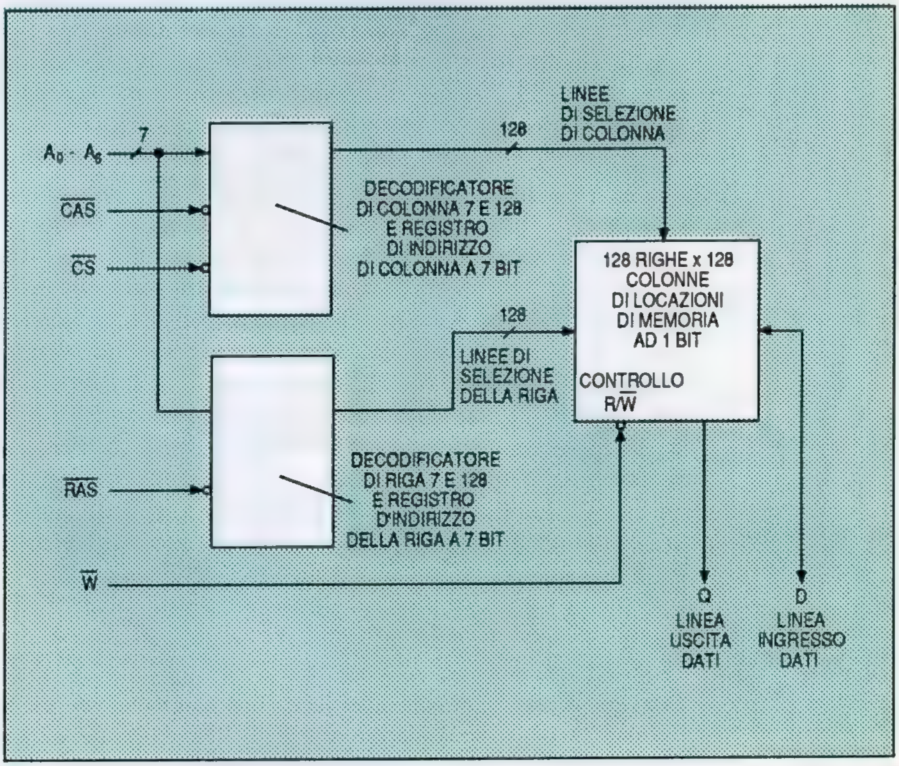

SELEZIONE DELLA RIGA
R/$\overline{W}$
SELEZIONE DELLA COLONNA
$\overline{CS}$
GRD
A TRF DELLE ALTRE CELLE NELLA STESSA RIGA
V_{DD}
SELEZIONE DELLA RIGA
R/$\overline{W}$
GRF
TRFL
BUS
Ripristino
TRFL
TRD
TL
V_{DD}
SELEZIONE DELLA RIGA
SELEZIONE DELLA COLONNA
R/$\overline{W}$
$\overline{CS}$
GW
TW
TS
CS
Leggi
Memorizza
Scrivi
V_{SS}
R/$\overline{W}$
$\overline{CS}$
Condensatore di immagazzinamento
UNA LINEA DATI
Buffer a 3 stati

L'elemento capacitivo CS collegato con la linea dati tramite un transistore di scrittura (TW) quando la linea $R/\overline{W}$ è bassa si carica al livello logico sul bus dati per una funzione di scrittura. Quando $R/\overline{W}$ è alto per una funzione di lettura, il livello logico di CS viene mandato al bus dati tramite un invertitore e un buffer a tre stati.

sto permette al condensatore di caricarsi al livello 0 o 1 presente sulla linea di ingresso dati per memorizzare i dati in ingresso. Se $R/\overline{W}$ è alto, si verifica un'operazione di lettura e si verificano due conseguenze. La prima è che il transistore TRD collega l'uscita del transistore di memoria TS con il buffer a tre stati per un'operazione di lettura quando la cella viene selezionata dagli ingressi della porta GRD. La tensione del condensatore viene invertita due volte: una volta dall'invertitore composto da TS, TRD e TL, e una volta dal buffer a tre stati il quale viene abilitato quando $R/\overline{W}$ è alto e $\overline{CS}$ è basso. Perciò l'uscita a livello 0 o 1 sulla linea dati sarà uguale a quella memorizzata in CS.

Nello stesso tempo ogni cella nella riga selezionata viene ripristinata collegando l'uscita del transistore TS tramite un invertitore e il transistore TRF al condensatore di memoria per rinnovarne la carica. Questo avviene in corrispondenza di un livello basso sull'uscita della porta GRF per la riga selezionata durante un'operazione di lettura. Perciò una riga di celle viene ripristinata quando si verifica una normale operazione di lettura. Una riga di celle può anche essere ripristinata tramite una lettura di ripristino nella quale $R/\overline{W}$ è alto, ma $\overline{CS}$ si trova nel suo stato inattivo. Sotto queste condizioni il dato nella cella non viene inviato sulla linea dati finché il buffer a tre stati non viene abilitato da $\overline{CS}$ e il transistore TRD non viene portato in conduzione. In tutti i casi il transistore TL è sempre inserito per agire come un resistore di cari-

Figura 6.5. RAM dinamica TMS 4116


$A_0 - A_6$
7
128
LINEE DI SELEZIONE DI COLONNA
$\overline{CAS}$
$\overline{CS}$
DECODIFICATORE DI COLONNA 7 E 128 E REGISTRO DI INDIRIZZO DI COLONNA A 7 BIT
128 RIGHE x 128 COLONNE DI LOCAZIONI DI MEMORIA AD 1 BIT
128
LINEE DI SELEZIONE DELLA RIGA
CONTROLLO $R/\overline{W}$
$\overline{RAS}$
DECODIFICATORE DI RIGA 7 E 128 E REGISTRO D'INDIRIZZO DELLA RIGA A 7 BIT
$\overline{W}$
Q
LINEA USCITA DATI
D
LINEA INGRESSO DATI

co o un resistore pull-up per le celle di memoria. Il transistore TRFL viene utilizzato per fornire l'azione di pull-up per un'operazione di solo ripristino.

La carica su ciascun elemento capacitivo CS, in una riga di celle, viene ripristinata ogniqualvolta quella riga è stata scelta per una funzione di lettura.

DRAM a circuito integrato

La singola cella di *figura 6.4* viene duplicata molte volte in un circuito integrato e le celle vengono organizzate in una matrice come indicato in *figura 6.5*. In questo tipo di struttura il numero delle righe e delle colonne solitamente è lo stesso. Il prodotto del numero delle righe per quello delle colonne è uguale al numero delle celle binarie di memorizzazione. Generalmente questi tipi di circuiti hanno 16.384 o 64.536 locazioni di memoria di 1 bit ciascuna. Si noti che ciascun bit si può indirizzare individualmente in questo circuito integrato, mentre nella RAM statica vista come esempio in precedenza se ne possono indirizzare otto alla volta.

Sottosistema di memoria

Per fornire un sottosistema di memoria con 8 bit per locazione indirizzabile devono essere collegati in parallelo otto circuiti di questo tipo, come mostrato per una memoria di 16.384 byte in *figura 6.6*, nella quale sono collegate in parallelo 8 RAM dinamiche del tipo TMS4116. Si noti che tutte le linee di indirizzo e di controllo di tutti e otto i circuiti integrati sono collegate fra loro. Comunque l'ingresso e l'uscita di ciascun circuito integrato vengono collegati con una linea separata del bus dati compresa tra D_0 e D_7 Il segnale di controllo $\overline{WE}$ che va all'ingresso $\overline{W}$ del circuito integrato determina se si sta svolgendo un'operazione di scrittura o di lettura. Se $\overline{W}$ viene portato al livello basso con dati sulle linee dati, i dati verranno scritti nella memoria. Se W è al livello alto, i dati verranno letti dalla memoria.

L'esempio di un sottosistema di memoria da 16 kilobyte che ha un formato di parola di 8 bit per locazione d'indirizzo ha una configurazione tale che tutti i bus d'indirizzo e di controllo sono fra loro collegati in parallelo: perciò essi lavorano simultaneamente. Le uscite, che sono indipendenti, rappresentano ciascuna un bit nella parola di 8 bit.

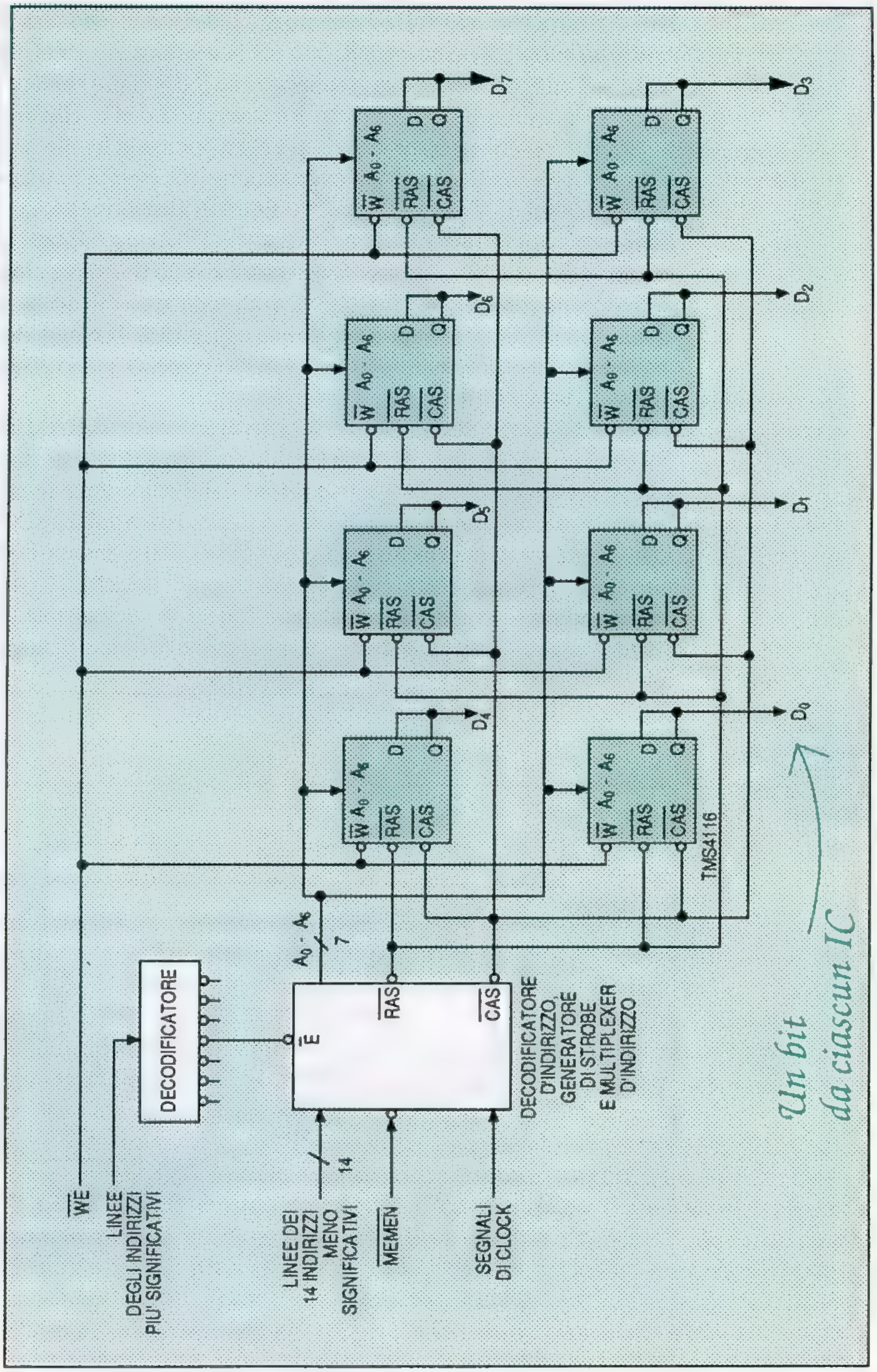
Lo scopo dei segnali $\overline{RAS}$ e $\overline{CAS}$ non è così semplice. Il segnale $\overline{RAS}$ viene utilizzato per convalidare l'indirizzo della riga, solitamente i sette bit di indirizzo meno significativi, nella memoria. Naturalmente l'indirizzo deve trovarsi sugli ingressi di indirizzo compresi tra A_0 ed A_6 del circuito di memoria nel momento in cui $\overline{RAS}$ compie la sua transizione dal livello alto al livello basso. Quindi devono essere collegate con le linee da A_0 ad A_6 le successive sette linee di indirizzo più significative e si deve portare il segnale $\overline{CAS}$ a compiere un transizione dal livello alto al livello basso. Tale transizione provocherà il verificarsi dell'operazione selezionata (lettura o scrittura). Se il segnale $\overline{CAS}$ rimane alto si verificherà un'operazione di ripristino per tutte le celle indirizzate della riga, ma non verrà emesso alcun dato perché $\overline{CAS}$ abilita il buffer di uscita a tre stati in ciascun circuito. Analogamente non si verificherà un'operazione di scrittura se il segnale $\overline{CAS}$ rimane alto.

Temporizzazione

Tutti questi segnali devono aver luogo con un preciso ordine temporale perché le operazioni vengano compiute esattamente. Il riparatore deve comprendere le relazioni temporali tra i segnali per trovare i guasti nelle memorie. Consideriamo allora la temporizzazione di una RAM statica.

Il diagramma di temporizzazione per il ciclo di lettura del circuito TMS4016 in *figu-*

Figura 6.6. Sottosistema di memoria dinamica composto da TMS4116


$\overline{WE}$
LINEE DEGLI INDIRIZZI PIU' SIGNIFICATIVI
DECODIFICATORE
LINEE DEI 14 INDIRIZZI MENO SIGNIFICATIVI
14
$\overline{MEMEN}$
SEGNALI DI CLOCK
$\overline{E}$
$\overline{RAS}$
$\overline{CAS}$
DECODIFICATORE D'INDIRIZZO, GENERATORE DI STROBE E MULTIPLEXER D'INDIRIZZO
$A_0 - A_6$
7
$\overline{W}$
$A_0 - A_6$
$\overline{RAS}$
$\overline{CAS}$
D
Q
TMS4116
D_0
D_1
D_2
D_3
D_4
D_5
D_6
D_7

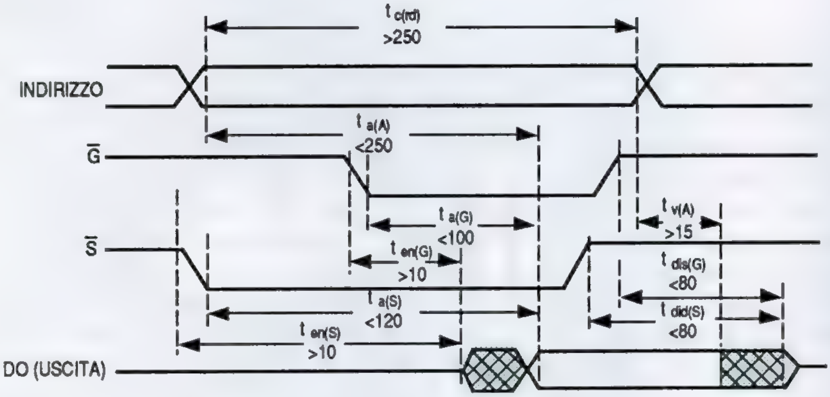
Un bit da ciascun IC

ra 6.7 mostra che, una volta stabilizzato l'indirizzo, il dato sarà pronto per poter essere letto entro 250 nanosecondi (ns) per la versione più lenta di questo circuito integrato. Tutto ciò avviene se si suppone che $\overline{S}$ e $\overline{G}$ siano nei loro stati attivi bassi. Il dato è disponibile 120 ns dopo che $\overline{S}$ è stato portato al livello basso. Analogamente il dato è disponibile 100 ns dopo che è stato portato al livello basso $\overline{G}$. Fintanto che il processore fornirà questa temporizzazione la memoria funzionerà correttamente. Le operazioni di scrittura richiedono limiti di temporizzazione simili. I segnali di temporizzazione di scrittura non sono mostrati in *figura 6.7*, ma i requisiti fondamentali sono che $\overline{S}$ e $\overline{W}$ siano mantenuti ad un livello basso per almeno 120 ns mentre il dato viene applicato al dispositivo, a cui sono applicati anche segnali di indirizzo stabili. Questi tempi vengono facilmente rispettati dalla maggior parte dei microprocessori moderni. Per microprocessori molto veloci ci sono versioni della TMS4016 che sono compatibili ed altrettanto veloci.

I requisiti di temporizzazione per il ciclo di lettura della RAM dinamica TMS4116 sono mostrati in *figura 6.8*. Nella versione puì lenta di questo dispositivo dovrebbero passare almeno 35 ns tra la transizione dal livello alto al livello basso del segnale $\overline{RAS}$ e la transizione dal livello alto al livello basso del segnale $\overline{CAS}$. Quindi $\overline{CAS}$ deve essere mantenuto basso per almeno 165 ns. Dal momento in cui $\overline{CAS}$ è stato portato al livello basso, il dato (DO) diventa disponibile entro 165 ns. Poi $\overline{RAS}$ deve essere riportato al livello alto per almeno 150 ns. Analogamente $\overline{CAS}$ deve essere riportato al livello alto per almeno 100 ns alla fine del ciclo di memoria. L'intera ope-

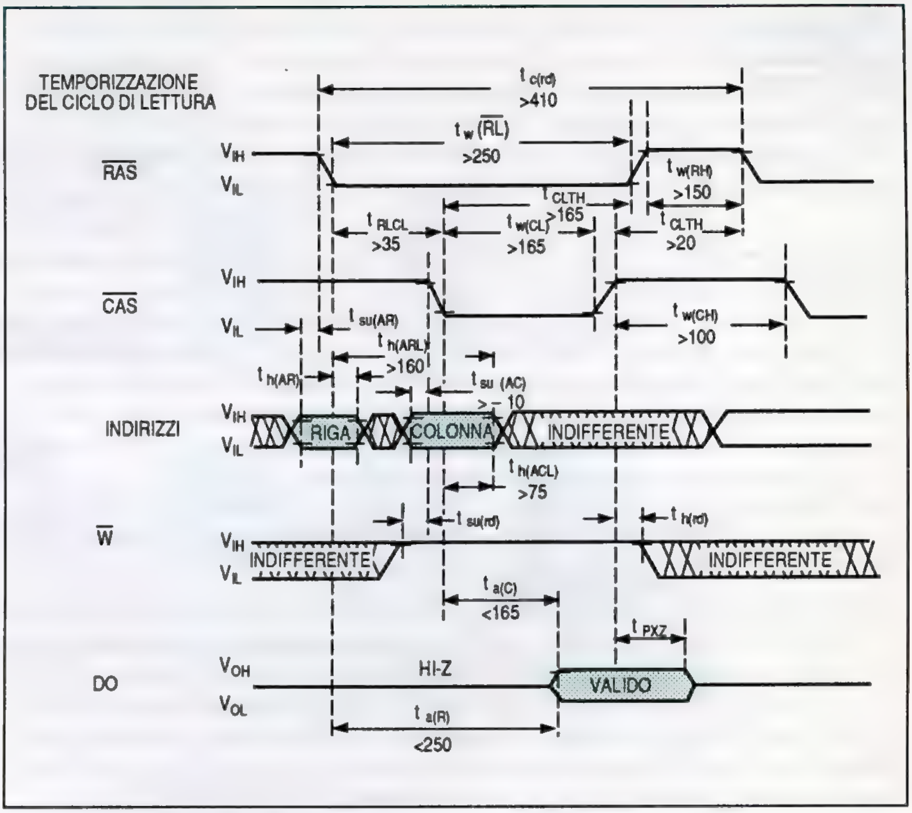
Poiché le funzioni digitali sono sequenziali e correlate le relazioni di tempificazione tra gli eventi, la durata degli eventi e i tempi di transizione tra lo stato 1 e lo stato 0 sono critici.

Figura 6.7. Diagramma temporale per un ciclo di lettura di un TMS4016



Tutti i punti di riferimento di tempificazione sono a 0,8 volt e 2,0 volt agli ingressi e 0,6 e 2,2 volt alle uscite (90%). I tempi di salita e discesa in ingresso sono uguali a 10 nanosecondi. Tutti i tempi sono espressi in nanosecondi.

Figura 6.8. Diagramma temporale per un ciclo di lettura di un TMS4116

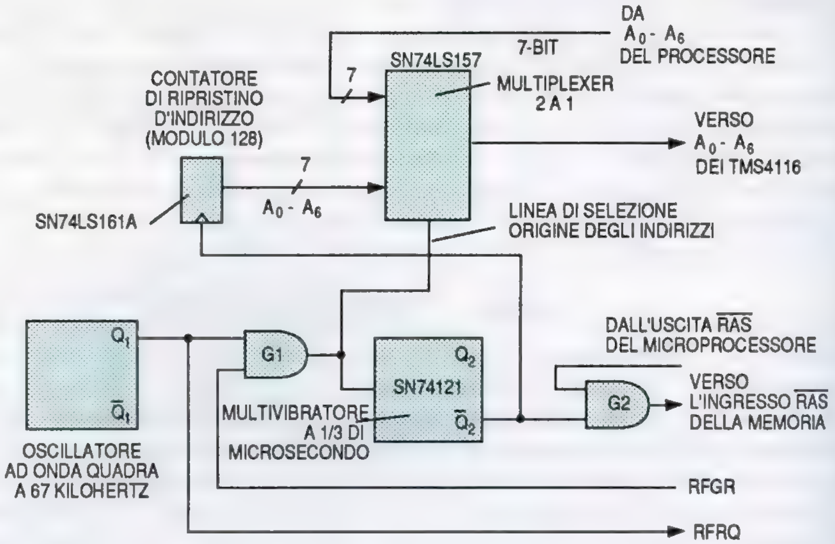

TEMPORIZZAZIONE
DEL CICLO DI LETTURA
t c(rd)
>410
t w (RL)
>250
RAS
V IH
V IL
t w(RH)
>150
t CLTH
>165
t RLCL
>35
t w(CL)
>165
t CLTH
>20
CAS
V IH
V IL
t su(AR)
t h(ARL)
>160
t h(AR)
t su (AC)
>= 10
t w(CH)
>100
INDIRIZZI
V IH
V IL
RIGA
COLONNA
INDIFFERENTE
t h(ACL)
>75
W
t su(rd)
t h(rd)
V IH
V IL
INDIFFERENTE
INDIFFERENTE
t a(C)
<165
t PXZ
DO
V OH
V OL
HI-Z
VALIDO
t a(R)
<250

Una RAM dinamica richiede un'operazione di ripristino ogni 2 millisecondi. Nell'esempio della memoria del circuito integrato il controllore di ripristino deve indirizzare ciascuna riga ogni 15 microsecondi per completare il ripristino totale in 2 millisecondi.

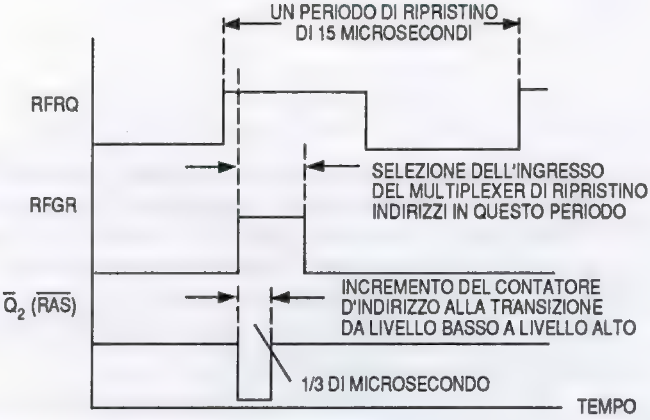
razione di lettura di memoria deve richiedere almeno 410 ns. Questi tempi vengono rispettati facilmente solo dai microprocessori di tipo più vecchio. Versioni più veloci della TMS4116 sono disponiili per i microprocessori più recenti (seconda e terza generazione) che operano a velocità maggiori (intervalli di clock più veloci). Per un'operazione di scrittura occorrono tempi simili. La temporizzazione di scrittura non è mostrata in *figura 6.8*, ma i dati di scrittura devono restare stabili e $\overline{\text{W}}$ deve rimanere basso per almeno 75 ns.

La RAM dinamica per funzionare correttamente richiede non solo che si forniscano corretti segnali di controllo di lettura e di scrittura e una corretta sequenza di segnali $\overline{\text{RAS}}$ e $\overline{\text{CAS}}$, ma anche i segnali di rinfresco, in quanto l'intera memoria deve essere ripristinata ogni 2 millisecondi (0,002 secondi). Ciò viene fatto fornendo alla memoria ogni indirizzo di riga con un segnale $\overline{\text{RAS}}$ quando il processore non sta accedendo alla memoria. Nel caso della TMS4116 ci sono 128 righe per 128 colonne con locazioni di memoria da 1 bit; perciò deve essere fornito un indirizzo di riga a sette bit al dispositivo ogni 15,6 microsecondi (0,002 secondi / 128 righe = 0,0000156 secondi o 15,6 microsecondi). L'indirizzo di riga deve essere incrementato dopo ogni operazione di ripristino cosicché ciascun ripristino operi su una riga diversa della me-

Figura 6.9. Controllore di ripristino di una RAM dinamica


DA
A_0 - A_6
DEL PROCESSORE
7-BIT
SN74LS157
MULTIPLEXER
2 A 1
7
CONTATORE
DI RIPRISTINO
D'INDIRIZZO
(MODULO 128)
VERSO
A_0 - A_6
DEI TMS4116
7
SN74LS161A
A_0 - A_6
LINEA DI SELEZIONE
ORIGINE DEGLI INDIRIZZI
DALL'USCITA $\overline{RAS}$
DEL MICROPROCESSORE
Q_1
$\overline{Q}_1$
G1
Q_2
SN74121
$\overline{Q}_2$
G2
VERSO
L'INGRESSO $\overline{RAS}$
DELLA MEMORIA
MULTIVIBRATORE
A 1/3 DI
MICROSECONDO
OSCILLATORE
AD ONDA QUADRA
A 67 KILOHERTZ
RFGR
RFRQ

a. Schema a blocchi



b. Diagramma temporale

moria. Perciò il ripristino interesserà le 128 righe, in modo che l'intera memoria venga ripistinata ogni 2 millisecondi. Nel circuito dell'unità di controllo del ripristino dovrà dunque esserci un temporizzatore a cui l'unità di controllo possa richiedere l'accesso alla memoria per un'operazione di ripristino ogni 15 microsecondi. Durante l'operazione, solo il segnale $\overline{RAS}$ dovrà essere portato ad un livello basso (non il segnale $\overline{CAS}$) per fornire un ripristino sicuro della memoria senza l'alterazione o la distruzione di alcun dato.

Un esempio di una semplice unità di controllo che potrebbe compiere questa operazione è mostrato in *figura 6.9*. Poiché $f = 1/T$, la tempificazione di 15 microsecondi viene fornita da un oscillatore astabile che produce un'onda quadra di 67kHz. Quando il terminale Q_1 dell'oscillatore ad onda quadra passa al livello logico 1, un segnale di richiesta di ripristino (RFRQ da refresh request signal) viene inviato al sistema di memoria. Il circuito deve rispondere con un segnale di ripristino accettato (RFGR da refresh grant signal) quando la memoria non sta per essere usata dal processore per un periodo che va da un minimo di 1/3 di microsecondo ad un massimo di 1/2 microsecondo. Il segnale RFGR passa attraverso una porta AND (G1) al terminale di clock del multivibratore da 1/3 di microsecondo per generare l'impulso $\overline{RAS}$. Si noti che, durante l'impulso RFGR, il contatore di ripristino di indirizzo (un contatore modulo 128) fornisce le uscite da A_0 ad A_6 che passando attraverso il multiplexer di indirizzo vengono portate sulle linee dalla A_0 alla A_6 del TMS4116.

Poi l'impulso $\overline{RAS}$ proveniente dal multivibratore passa attraverso la porta AND $\overline{RAS}$ (G2) per inviare il segnale $\overline{RAS}$ al TMS4116 in modo tale da provocare il verificarsi del ripristino. Si noti che non viene generato alcun segnale $\overline{CAS}$. La transizione dal livello basso al livello alto del segnale di ripristino $\overline{RAS}$ da $\overline{Q}_2$ viene utilizzata per incrementare il contatore di indirizzo in modo tale che sia pronto l'indirizzo per il successivo ciclo di ripristino. Il multiplexer di indirizzo viene commutato nuovamente alle linee di indirizzo del processore per normali operazioni di memoria quando RFGR ritorna allo stato basso. Questa sequenza si ripeterà ogni 15 microsecondi ogni volta che l'uscita astabile RFRQ va al livello alto. Poiché tutti i segnali nel circuito di *figura 6.9* cambiano ad una frequenza di circa 67 kilohertz, la maggior parte delle tecniche di ricerca ed eliminazione dei guasti trattate nei primi capitoli di questo libro non è sufficiente a determinare se l'unità di controllo di ripristino stia operando correttamente o no.

Infatti la maggior parte delle tecniche trattate non permetterebbe al riparatore di verificare i requisiti di temporizzazione mostrati nelle *figure 6.7* e *6.8*. Comunque la maggior parte delle misurazioni di resistenza e in corrente continua discusse nei capitoli precedenti dovrebbe aiutare il riparatore a determinare se un circuito di memoria è guasto. Per guasti in cui si abbia corto circuito o circuito aperto, la RAM dinamica, la RAM statica e le memorie a sola lettura saranno analizzate in modi simili. La principale differenza esterna tra RAM e ROM è che devono essere verificate entrambe le operazioni di lettura e scrittura nelle RAM, mentre, per le ROM, devono essere esaminate solo le operazioni di lettura. Tecniche più avanzate che permettono la verifica dei segnali di temporizzazione che potrebbero essere necessarie per alcuni problemi di memoria verranno discusse nel capitolo 8. Alcune di queste tecniche sono necessarie per determinare se una memoria dinamica è stata ripristinata in modo appropriato.

PROBLEMI DEI COMPONENTI DELLA MEMORIA

Poiché vi è un numero molto elevato di locazioni di memoria, per verificare lo stato in cui si trovano i decodificatori, le porte e la memoria stessa deve essere utilizzato un test di memoria basato sul computer.

I componenti di un sottosistema di memoria comprendono decodificatori, porte, RAM e ROM. In *figura 6.3* è mostrato un tipico uso di questi componenti in un sottosistema di memoria RAM. Altre memorie userebbero questi componenti in maniera simile. In termini di numero di circuiti integrati diversi, la memoria di *figura 6.3* non è così complicata come un sistema di macchina sequenziale del tipo considerato nel capitolo 5. Ciò che rende la memoria più difficile al riparatore è il gran numero di locazioni di memorizzazione diverse per cui deve essere verificato il funzionamento. Un singolo circuito integrato RAM può avere da 2048 a 65.536 locazioni interne e molti sottosistemi di memoria possiedono più di un circuito integrato RAM. È un compito impossibile provare e verificare manualmente lo stato operativo di ogni locazione. Perciò, se il problema è all'interno di un circuito integrato RAM, si dovranno utilizzare alcuni tipi di dispositivi di test delle memorie, probabilmente basati su di un microcomputer, per determinare se un dato circuito di memoria funziona o no. Comunque prima di provare il circuito di memoria stesso si possono verificare i circuiti combinatori più semplici (decodificatori e porte) in maniera relativamente facile e veloce.

Guasti nel circuito combinatorio

L'isolamento dei componenti difettosi del decodificatore comporta l'immissione di una sequenza di segnali nota e il controllo dell'uscita con un'apposita strumentazione di test. Per quanto riguarda i dispositivi fondamentali, le porte, esse vengono impulsate e si controllano le uscite che ne risultano.

Nel circuito di figura 6.3 una porta OR genera l'abilitazione per un semplice decodificatore 2 a 4 dal segnale $\overline{\text{MEMEN}}$ e dal bit di indirizzo più significativo. Si può verificare il funzionamento del decodificatore facendolo lavorare con segnali applicati alle tre linee di indirizzo più significative. Questi segnali potrebbero provenire dal microprocessore del sistema o da uno strumento atto a provare la memoria. La linea di indirizzo più significativa dovrebbe alternarsi tra i valori 0 ed 1, assumendo il valore 0 per accedere alla memoria al fine di leggere o scrivere. Si deve avere a disposizione uno strumento di prova sufficientemente veloce per controllare i segnali. (Tale tipo di strumento verrà trattato nel capitolo 8.) Con questi segnali di prova applicati l'uscita della porta OR dovrebbe alternarsi tra il valore 0 (operazione di memoria) e il valore 1 (operazione non-RAM). Inoltre i segnali sulle linee di selezione del decodificatore dovrebbero indicare lo stesso codice di selezione a 2 bit per operazioni alternate. Queste verifiche dovrebbero indicare se il circuito dal processore al decodificatore funziona correttamente.

Se l'uscita della porta OR resta sempre fissa a 0 o ad 1, oppure ciò succede ad una o più linee di selezione, il guasto deve trovarsi sulla linea tra il processore e gli ingressi o di selezione del decodificatore o della porta OR, oppure deve trovarsi nella porta OR stessa. Il corretto funzionamento di queste linee e porte può essere determinato rimuovendo il microprocessore dal sistema e utilizzando tecniche di rilevazione del segnale (misurazioni con sonda logica e impulsatore o con voltmetro) lungo le linee in questione. Analogamente il funzionamento della porta OR può essere verificato isolando i soli piedini di ingresso e il suo piedino di uscita dal processore e dal decodificatore e applicando condizioni di ingresso note in modo tale da controllare se si verifica l'uscita corretta.

Una volta determinato che i segnali nel decodificatore funzionano correttamente, l'apparecchiatura di prova ad alta velocità dovrebbe essere usata per verificare che l'uscita del decodificatore selezionata (e non altre) si alterna tra i valori 0 ed 1. Se ciò non si verifica (cioè se si verifica che l'uscita del decodificatore è fissa a 0 o ad 1), allora o il decodificatore è guasto o c'è un guasto nel cablaggio tra l'uscita del decodificatore e gli ingressi di selezione del chip di memoria. Anche in questo caso possono essere usate tecniche di isolamento dei piedini e di rilevazione della traccia per determinare la natura del guasto e le correzioni da apportare. Tali correzioni possono consistere sia nel sostituire il circuito del decodificatore che nel riparare il cablaggio. Il funzionamento del decodificatore dovrebbe essere verificato per tutti i possibili codici di selezione (lavorando correttamente le linee di uscita del decodificatore dalla 0 alla 3).

Il funzionamento del decodificatore e della porta OR può essere verificato manualmente immettendo combinazioni di tutti 1 e di tutti 0.

Se non fosse possibile verificare il funzionamento del decodificatore e della porta OR alle velocità di un normale microprocessore (solitamente al di sopra di 1 megahertz), è possibile verificare il funzionamento di questi dispositivi con le tecniche presentate nei capitoli dal 2 al 4. Dapprima le tre linee di indirizzo più significative e la linea $\overline{\text{MEMEN}}$ verso il microprocessore devono essere isolate o aperte. Ciò può essere fatto rimuovendo il circuito integrato del microprocessore se questo si trova su di uno zoccolo. Quindi si pongano degli 0 logici sulle tre linee di indirizzo e sulla linea $\overline{\text{MEMEN}}$. Misurazioni di tensione dovrebbero verificare le presenze di 0 logici su tutti gli ingressi del decodificatore e solo sull'uscita 0 del decodificatore stesso. Le altre linee di uscita del decodificatore potranno essere verificate mettendo manualmente tensioni di 0 ed 1 logici sulle linee di selezione del decodificatore secondo tutte le possibili combinazioni, mentre l'uscita della porta OR viene mantenuta al livello 0. Se la porta OR o il decodificatore non funzionano correttamente, il circuito o il cablaggio guasto può essere riparato o sostituito.

Naturalmente nessuna di queste verifiche andrebbe fatta a meno che non si sospetti o non si sappia per certo che esiste un problema nel sottosistema di memoria. Infatti non è cosa molto semplice circoscrivere il guasto di un microcomputer alla memoria ed è anche più difficile isolarlo in un particolare sottosistema di memoria. Inoltre, anche se è noto che il guasto è localizzato nella memoria, le tecniche trattate finora in questo capitolo potrebbero indicare solo se il guasto si trova o no nella parte combinatoria della memoria (il decodificatore e le porte). Se il guasto non è situato in questi dispositivi è certamente molto più difficile determinare quale circuito di memoria è guasto e perché. Alcuni dei problemi nel determinare quale circuito di memoria è guasto verranno considerati nei prossimi paragrafi.

Guasti nei circuiti di memoria

Nel normale funzionamento di un sottosistema di memoria, sia esso ROM o RAM, non si può accedere ad una locazione guasta che molto tempo dopo che il sistema è stato in funzione. Ciò si verifica soprattutto se il problema riguarda una parte dei circuiti della RAM. La sola indicazione per un tale tipo di problema potrebbe essere un errore occasionale nei dati di un sistema di elaborazione dati, oppure un errore occasionale nella posizione di una parte della macchina in un sistema di controllo di una macchina. Un tale errore potrebbe essere dovuto a problemi di ingresso o uscita oppure ad un processore guasto, oltre che alla memoria stessa. Infatti un malfunziona-

La verifica del funzionamento della memoria richiede tecniche automatiche in quanto sono coinvolte moltissime locazioni. La verifica automatica viene svolta da strumentazione di test o da un programma già inserito nella memoria.

mento di un qualsiasi circuito o filo del sistema potrebbe causare tale problema. Se una ROM è difettosa e il programma del sistema è memorizzato in quella ROM, allora il sistema non funzionerà correttamente o smetterà di funzionare del tutto. Comunque risposte non attendibili a comandi dati tramite la tastiera potrebbero essere dovute ad una tastiera difettosa o a problemi di ingresso. Ancora una volta, quindi, può non essere semplice circoscrivere il problema con certezza alla sola memoria del sistema.

Verifiche del funzionamento di una memoria

Anche se è certo che la memoria è guasta, deve essere possibile provare automaticamente la memoria locazione per locazione. La verifica manuale del funzionamento delle locazioni (da 2000 a 64.000) applicando codici di indirizzo successivi e segnali di tempificazione sulla memoria per leggere poi i contenuti di ciascuna locazione richiederebbe centinaia o migliaia di ore di lavoro. Ovviamente questo non può

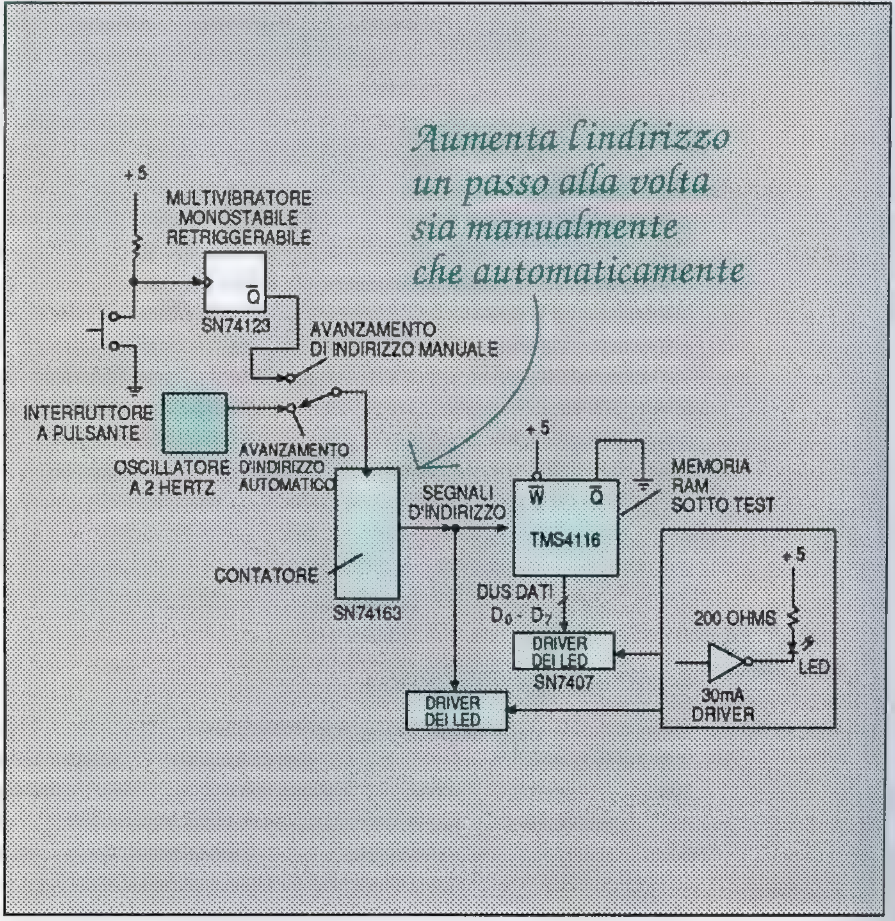

Aumenta l'indirizzo un passo alla volta sia manualmente che automaticamente
+5
MULTIVIBRATORE MONOSTABILE RETRIGGERABILE
Q̄
SN74123
AVANZAMENTO DI INDIRIZZO MANUALE
INTERRUTTORE A PULSANTE
OSCILLATORE A 2 HERTZ
AVANZAMENTO D'INDIRIZZO AUTOMATICO
CONTATORE
SN74163
SEGNALI D'INDIRIZZO
+5
W̄
Q̄
TMS4116
MEMORIA RAM SOTTO TEST
DUS DATI D0 - D7
DRIVER DEI LED
SN7407
DRIVER DEI LED
+5
200 OHMS
LED
30mA DRIVER

Figura 6.10. Semplice circuito di test di memoria

essere accettabile. La verifica automatica del funzionamento della memoria solitamente viene fatta collegando uno speciale strumento di verifica o facendo eseguire al processore del sistema un programma per la verifica del funzionamento della memoria. Infatti, in alcuni sistemi, un programma per la verifica del funzionamento della memoria gira automaticamente ogniqualvolta il sistema viene acceso. Solitamente i risultati di una sequenza di verifica automatica appaiono sul display di un CRT o vengono stampati con una stampante per informare l'operatore di quale circuito sia guasto e perché.

Una cella di memoria guasta può essere isolata immettendo in memoria, in corrispondenza d'intervalli periodici, la completa sequenza d'indirizzi ed individuando una configurazione data e memorizzata di dati. I segnali d'indirizzo possono essere generati da contatori o da un microprocessore.

Se tali strumenti o programmi di verifica non sono inclusi nel sistema, ci sono alcune semplici tecniche che possono essere utilizzate per determinare un circuito guasto nel caso in cui il processore possa essere costretto a fornire segnali di indirizzo che passino ordinatamente attraverso locazioni di memoria successive e sia diponibile uno strumento di verifica sufficientemente veloce. Una tecnica esemplificativa è mostrata in *figura 6.10.* Il processore è sostituito da un contatore che genera segnali di indirizzo per indirizzi successivi. Un interruttore a due posizioni permette di utilizzare due modi differenti per incrementare l'indirizzo. Nella posizione manuale l'indirizzo viene incrementato solo quando il pulsante è premuto. Ogni volta che si preme il pulsante l'indirizzo aumenta di 1. Nella posizione automatica, l'uscita dell'oscillatore fornisce un clock molto lento cosicché l'indirizzo viene continuamente aumentato di 1 ogni mezzo secondo circa. Gli ingressi di controllo vengono predisposti manualmente per abilitare il circuito di memoria di cui si deve verificare il funzionamento per un'operazione di lettura. L'ingresso di indirizzo e le uscite dati della memoria vengono controllate da alcuni LED nel circuito di verifica di *figura 6.10* cosicché i livelli logici 0 ed 1 di tutte le uscite della memoria possono essere rilevati contemporaneamene.

Nell'esempio i LED mostreranno solo i più gravi difetti della memoria tramite la combinazione delle luci che si verificherà durante la sequenza di test.

Se una data memoria presenta una condizione di 0 fisso o di 1 fisso in una data posizione di uscita (o in più posizioni di uscita), questo sarà indicato da un LED che resta sempre acceso o spento mentre i LED sulle altre uscite o sulle linee di indirizzo cambiano stato. I LED di indirizzo presenteranno una configurazione di conteggio binaria mentre i LED sull'uscita della memoria dovrebbero accendersi e spegnersi in maniera casuale. Se un qualunque LED di uscita dovesse rimanere spento o acceso per tutti gli indirizzi, la RAM sarà guasta internamente per quell'uscita dati o ci sarà un guasto nel cablaggio o nei collegamenti. Questa sequenza di prove dovrebbe essere svolta su ciascuna delle altre ROM o RAM nel sottosistema di memoria. Questa verifica individuerà solo i guasti più evidenti, ma può essere svolta con un circuito di prova relativamente poco costoso. Se il contatore incrementa alla frequenza di 2 hertz, cosicché gli indirizzi vengono cambiati ogni mezzo secondo, occorrono 4096 secondi (più di 1 ora) per verificare una memoria a 8192 byte. Si può verificare una memoria alla frequenza di 1 megahertz se è possibile utilizzare uno strumento controllato da un processore o se il sistema ha una memoria con la caratteristica di possedere un programma di autotest inserito nel programma del sistema. A questa frequenza di 1 MHz si potrà verificare il funzionamento di una memoria da 8192 byte in meno di 9 millisecondi, tempo che non è rilevabile da un operatore. Anche una memoria da 64 kilobyte può essere verificata in meno di 66 millisecondi, mentre con le tecniche mostrate in *figura 6.10* occorrerebbero circa 9 ore. Nei prossimi paragrafi verranno considerate verifiche automatiche di memoria.

PROCEDURE PER LA VERIFICA DEL FUNZIONAMENTO

La cosa fondamentale di un test di memoria è l'essere in grado di verificare che ciascuna locazione è in grado di memorizzare e di leggere uno stato 1 o uno stato 0. La strumentazione per il test deve essere in grado di isolare l'area difettosa e di dare informazioni al tecnico.

L'obiettivo di un programma per la verifica del funzionamento di una RAM è quello di provare tutte le celle di memoria del sistema per controllare la loro capacità di memorizzare e leggere 0 ed 1. Lo strumento o il programma di prova deve semplicemente scrivere uno 0 in ciascuna cella e leggere quella cella per verificare la presenza dello 0, quindi scrivere un 1 in ciascuna cella e di nuovo leggere per verificare la presenza dell'1. L'obiettivo fondamentale di un programma di prova per una ROM è quello di verificare che vi siano memorizzate le sequenze di codici desiderate nella ROM. Le sequenze di codici desiderate devono anche essere conosciute dallo strumento di prova, cosicché il codice che viene letto da ciascuna locazione nella ROM in prova possa essere confrontato con il codice desiderato. Se una locazione all'interno di una ROM o di una RAM viene trovata guasta, lo strumento di prova deve notificare all'operatore quale circuito è guasto e di che tipo di problema si tratta. Lo strumento può svolgere questo compito stampando un messaggio con una stampante, facendolo apparire sul display del CRT oppure accendendo un LED vicino al chip guasto. Il circuito guasto dovrebbe essere sostituito o provato ulteriormente con le tecniche descritte precedentemente per individuare le porte di controllo e i decodificatori guasti.

Prova di RAM statiche

Il programma di test può essere originariamente sia nella memoria del computer che nello strumento per il test.

Il principale programma di prova di una RAM per un processore di tipo TMS9980 è mostrato in *figura 6.11* insieme alla spiegazione del funzionamento di ciascuna istruzione. Nel primo caso questo programma risiederebbe nella memoria del sistema di cui si sta verificando il funzionamento. Il programma potrebbe essere eseguito quando il sistema è acceso per la prima volta per verificare il funzionamento della memoria oppure potrebbe essere eseguito in risposta ad un comando di prova della memoria. Per il secondo caso il programma risiederebbe nella memoria dello strumento di prova pilotato dal processore e che è collegato con il sistema. Esso disattiverebbe il processore dal sistema e controllerebbe le linee dati, le linee di temporizzazione e le linee di indirizzo. In entrambi i casi, il programma di prova della memoria eseguirebbe una sequenza di prove per verificare il funzionamento di una RAM da 2048 byte uguale a quella vista per la prima RAM di *figura 6.3*.

Prima che il programma principale chiami la subroutine TEST di *figura 6.11*, esso deve portare il registro 1 a 0, inizializzare i registri 6 e 7 agli esatti indirizzi iniziali per le tabelle FAULT, e azzerare la posizione chiamata FLAG. Fatto ciò, il programma principale chiama la subroutine TEST.

La prima istruzione della subroutine TEST porta il registro 2 a tutti 0 così che l'indirizzo dell'inizio del test è il numero esadecimale 0000. Quindi il codice binario di prova formato da 1 e 0 alternati, 0101 0101 0101 0101 (5555 esadecimale), viene caricato nel registro 3. Il valore 2048 è caricato nel registro 4, che serve come contatore. Ogniqualvolta viene testato un byte di memoria, il suo valore diminuisce di 1; perciò quando il suo valore è 0, tutti i 2048 byte saranno stati indirizzati. A questo punto l'istruzione MOV copia il dato esadecimale 5555 del registro 3 nell'indirizzo di memoria specificato nel registro 2; quindi incrementa l'indirizzo nel registro 2 per

Figura 6.11. Programma di test della RAM

```
TEST    CLR 2        Imposta R2 al primo indirizzo della memoria sotto test.
        LI 3,>5555   Imposta R3 a 0 e 1 alternati.
        LI 4,2048    Imposta R4 al conto 2048.
LOOP    MOV 3,*2+    Scrivi il codice di prova della RAM (Vedi Nota).
        DEC 4        Fai decrescere il contatore di byte.
        JNE LOOP     Se tutta la RAM non è scritta, continua a scriverea loop.
        CLR 2        Altrimenti imposta ancora R2 al primo indirizzo della memoria sotto test.
        LI 4,2048    Carica ancora il contatore a 2048.
LOP     MOV *+2,5    Leggi la memoria.
        XOR 3,5      Controlla se i bit letti sono gli stessi bit memorizzati.
        JNZ FAULT    Se non è così vai a FAULT,
CONT    DEC 4        Altrimenti decrementa il contatore.
        JNZ LOP      Se il contatore non è 0 continua a provare a loop,
        JMP NEXT     Altrimenti vai all'istruzione successiva.
FAULT   SETO FLAG    Se c'è un guasto, porta flag a tutti 1.
        MOV 2,*6+    Memorizza l'indirizzo del guasto.
        MOV 5,*7+    Memorizza la sequenza errata di bit.
        JMP CONT     Continua la prova.
NEXT    CI 1,1       La memoria è stata completamente controllata?
        JEQ OUT      Se si, vai ad OUT.
        INC 1        Altrimenti indica l'ultimo passo compiuto nel test.
        INV 3        Scambia le posizioni degli 0 e degli 1 nel codice di test.
        CLR 2        Imposta ancora R2 al primo indirizzo della memoria sotto test.
        LI 4,2048    Carica ancora il contatore a 2048.
        JMP LOOP     Continua la prova a loop con il nuovo codice di test.
OUT     RTWP         Ritorna al programma.
```

Nota: il segno positivo (+) dopo il registro indica che l'indirizzo verrà incrementato per evidenziare la locazione successiva cui si accederà dopo questa operazione.

accedere alla locazione successiva. Questo ciclo (loop) continuerà a scrivere il codice 5555 finché il registro 4 (il contatore) è 0, il che indica che tutte le 2048 locazioni sono state occupate dal codice di prova.

Il test reale viene compiuto nella seconda sequenza di istruzioni di lettura a cominciare da CLR2. Anche in questo caso l'indirizzo di 0000 esadecimale è caricato nel registro 2 e nel registro 4 viene messo il valore iniziale del contatore, cioè 2048. La posizione di memoria viene letta e il suo contenuto viene messo nel registro 5. I contenuti del registro 5 vengono confrontati con quelli del registro 3 (il quale contiene il codice di prova) per vedere se la sequenza dei bit è identica. Questo confronto viene compiuto con un'istruzione OR-Esclusiva (XOR) che metterà tutti 0 nel registro 5 se dalla posizione di memoria è stato letto il codice corretto. Se si otterranno tutti 0 il test continuerà normalmente. Se così invece non fosse, il programma salterebbe all'istruzione contrassegnata con l'etichetta FAULT.

Nella sequenza FAULT la posizione denominata con FLAG viene portata a tutti 1. Quindi l'indirizzo nel registro 2, che è l'indirizzo della locazione di memoria guasta, viene memorizzato in una tabella indirizzata dal registro 6. Dopo di che i contenuti del registro 5, il quale contiene un 1 in ciascuna posizione di bit non coincidente, vengono memorizzati in una tabella indirizzata dal registro 7. Dopo che le condizioni di guasto sono state registrate nelle tabelle, la prova continuerà tornando all'istru-

Questo programma di test per una RAM svolge le operazioni richieste per far girare un completo test di memoria.

zione contrassegnata con l'etichetta CONT. La fine di questa sequenza di prova si verificherà quando il contatore nel registro 4 sarà diminuito fino a diventare 0.

A questo punto si verifica un salto all'istruzione contrassegnata con l'etichetta NEXT e i contenuti del registro 1 vengono confrontati col numero 1. Se il registro 1 contiene 1, la prova è terminata e viene effettuato un salto all'istruzione OUT per ritornare dalla subroutine TEST al programma principale. Se il registro 1 contiene ancora 0, le verifiche non sono finite: perciò il registro 1 viene incrementato ad 1 e il codice di prova viene cambiato in 1010 1010 1010 1010 (AAAA esadecimale) tramite l'istruzione INV3. Si noti che questa configurazione è ancora formata da 1 e 0 alternati, ma le posizioni degli 0 e degli 1 sono state scambiate rispetto a quelle del primo codice di prova. Il registro 2 viene portato nuovamente a 0000, il registro 4 (il contatore) è caricato ancora con 2048 e viene effettuato un salto all'istruzione contrassegnata con l'etichetta LOOP per cominciare il nuovo test. Quando la subroutine TEST sarà stata completamente eseguita, cosa che verrà indicata da un 1 nel registro 1, il programma principale controllerà la posizione FLAG.

Se FLAG è formata da tutti 0, il circuito di memoria funziona correttamente e il programma principale può mandare un messaggio alla stampante o al display CRT che dice "La RAM 0 funziona correttamente". Quindi la subroutine TEST potrà continuare ad eseguire programmi simili per le tre successive RAM di *figura 6.3*. Infatti questa subroutine potrebbe provare tutte le RAM se il programma principale passasse l'indirizzo di inizio della RAM di cui deve essere verificato il funzionamento nel registro 2 tramite la variabile START. Naturalmente le istruzioni CLR2 dovrebbero essere cambiate in LI2, START.

Le scritture diagnostiche che vengono stampate informano il riparatore su quale pacchetto di memoria abbia un guasto e debba essere sostituito. Esse potrebbero fornire anche informazioni diagnostiche per isolare la parola e la locazione che è in errore.

Se dopo aver fatto girare la subroutine TEST la posizione FLAG contiene tutti 1 il programma principale eseguirà una sequenza di programma per stampare il messaggio "La RAM 0 è guasta e deve essere sostituita". Esso potrebbe anche stampare il contenuto delle tabelle dei guasti indirizzate dai registri 6 e 7 con una sequenza di messaggi del tipo: "La locazione (contenuti della tabella indirizzata da R6) è guasta nella posizione di bit (contenuti della tabella indirizzata da R7) per ogni accesso alle due tabelle". Tale messaggio stampato permetterebbe all'operatore di individuare con precisione quali locazioni e quali linee dati per queste locazioni (le posizioni di bit che contengono un 1) sono difettose. Se sono guaste solamente posizioni che dovrebbero avere memorizzato uno 0, la posizione di memorizzazione sarà bloccata ad 1. Analogamente se soltanto le posizioni che dovrebbero aver memorizzato un 1 sono guaste la posizione di memorizzazione sarà bloccata a 0. Ciò dovrebbe permettere al riparatore di utilizzare le tecniche di isolamento dei piedini per determinare se il cablaggio esterno è in corto circuito con il positivo dell'alimentatore o a massa, prima di sostituire un circuito di memoria che potrebbe funzionare correttamente.

Potrebbero essere aggiunte al programma alcune istruzioni in modo che il programma stesso possa determinare se un piedino è bloccato a 0 o ad 1 solo per alcune locazioni di memoria (indirizzi) o per tutte le locazioni di memoria. Se solo alcune locazioni sono bloccate a 0 o ad 1 allora è molto probabile che il circuito sia guasto. Se tutte le locazioni in una RAM presentano lo stesso piedino bloccato a 0 o ad 1, molto probabilmente c'è un corto circuito o un circuito aperto nel cablaggio esterno, sebbene una RAM guasta potrebbe lo stesso causare questo problema. Perciò se vengono aggiunte più istruzioni al programma diagnostico in modo tale che esso svolga più prove in maniera più dettagliata, il computer o lo strumento di prova potranno raccogliere molti dati la cui analisi potrà far determinare il guasto. Quindi esso potrà stampare un resoconto dettagliato per il riparatore. Un tale programma potrebbe fare praticamente tutto eccetto che riparare il guasto concretamente.

Verifica del funzionamento di una ROM

Le memorie ROM, PROM e EPROM possono essere controllate nello stesso modo con cui viene controllata una RAM.

I codici di programma o gli altri codici che si suppongono memorizzati nella ROM di cui deve essere verificato il funzionamento devono essere duplicati sullo strumento di prova. Esso confronta il suo codice di riferimento con quello della ROM. Una perfetta coincidenza fra i due codici indicherebbe che la ROM controllata funziona correttamente, mentre ogni differenza significherebbe che la ROM o il suo cablaggio sono guasti.

Lo schema di *figura 6.3* potrebbe essere convertito in una versione EPROM sostituendo direttamente tutti i componenti TMS4016 con i loro equivalenti EPROM, cioè i TMS2516. Si dovrebbe quindi collegare un sistema microcomputerizzato di prova per disattivare e sostituire il processore nel sistema di *figura 6.3*: in questo modo lo strumento di prova controllerebbe completamente il sottosistema EPROM. Una copia dei codici memorizzati nel sottosistema EPROM da 8192 byte e un programma di prova del tipo mostrato in *figura 6.12* dovrebbero essere memorizzati nello strumento di prova. Si supponga ché il codice da confrontare si trovi nella memoria dello strumento di prova nelle locazioni comprese fra 8000 e 9FFF (esadecimali) e che il programma per controllare la sequenza di prova (*figura 6.12*) sia messo nel-

Figura 6.12. Programma di test di una ROM

```
ROMTEST   LI 4,8192    Imposta il contatore a 8192.
          LI 3,>8000   Imposta l'indirizzo del file di riferimento a 8000.
          CLR 2        Imposta l'indirizzo di memoria sotto test a tutti 0.
LOOP      MOV *3+,5    Leggi il codice di riferimento a R5 (Vedi Nota).
          XOR *2+,5    Leggi la ROM sotto test e confronta col codice di riferimento.
          JNZ FAULT    Se non è uguale, vai a FAULT,
CONT      DEC 4        Altrimenti decrementa il contatore.
          JNZ LOOP     Se il contatore non è 0, continua la prova,
          RTWP         Altrimenti return.
FAULT     SETO FLAG    Se guasto, porta flag a tutti 1.
          MOV 2,*6+    Memorizza l'indirizzo guasto.
          MOV 5,*7+    Memorizza la sequenza di bit sbagliata.
          JMP CONT     Continua la prova.
```

Nota: Il segno positivo (+) indica che l'indirizzo verrà incrementato per indicare la locazione successiva cui si accederà dopo questa operazione.

le locazioni a partire da A000 (esadecimale). Lo strumento di prova può quindi avere accesso alla EPROM della *figura 6.3* modificata come se occupasse le posizioni dalla 0000 alla 1FFF (esadecimali) nello spazio riservato alla memoria dello strumento di prova. Sotto queste condizioni il programma di prova di *figura 6.12* sarebbe in grado di compiere il test desiderato.

Nel programma di *figura 6.12*, i registri 6 e 7 punteranno anche in questo caso agli indirizzi delle tabelle per memorizzare gli indirizzi delle locazioni di memoria guaste e le posizioni in cui ci sono dei bit errati. Il registro 4 funge ancora da contatore, ma in questo programma esso viene caricato con 8192 perché i byte di memoria da controllare sono esattamente 8192. Nel registro 2 verrà conservato ancora l'indirizzo del byte di memoria di cui deve essere verificato il funzionamento: perciò esso viene inizializzato a 0000. L'uso del registro 3 è invece differente in questo programma; esso punta all'indirizzo del byte del codice di riferimento nella memoria dello strumento di prova, byte che deve essere confrontato con il byte del codice EPROM indirizzato dal registro 2. Perciò il registro 3 è inizializzato con l'indirizo 8000 (esadecimale).

Quindi, iniziando dall'istruzione contrassegnata dall'etichetta LOOP, il byte di riferimento viene portato nel registro 5 e il byte proveniente dalla memoria EPROM sotto test è confrontato con questo byte di riferimento tramite un OR-Esclusivo (XOR). Un risultato di tutti 0 indica che la memoria provata funziona e il test continua con l'istruzione contrassegnata dall'etichetta CONT. Un 1 in una qualsiasi posizione di bit nel registro 5 dopo lo XOR indica un errore nell'EPROM sotto test e il programma quindi salta all'istruzione contrassegnata con FAULT. Questa sequenza è esattamente uguale a quella in *figura 6.12*.

Dopo che sono state registrate le condizioni di guasto la sequenza di prova continua ripartendo dall'istruzione etichettata CONT. A questo punto del programma il valore del contatore nel registro 4 sarà diminuito; si dovrà allora controllare se questo valore è 0 per essere sicuri che tutti gli 8192 byte sono stati controllati. Se così non è, il test continua con l'istruzione LOOP. Quando tutti i byte sono stati controllati si ritorna al programma principale del sistema. Il modo in cui il programma principale chiama la subroutine ROMTEST e tratta le condizioni di errore è uguale a quello spiegato per la *figura 6.11*.

Verifica del funzionamento di una RAM dinamica

La DRAM, sebbene abbia requisiti per il test simili a quelli di una RAM, presenta molte importanti differenze come l'organizzazione delle celle, i requisiti di ripristino delle celle e le relazioni tra i

La prova di una RAM dinamica è simile a quella utilizzata per una RAM statica. Le principali differenze riguardano i requisiti per le operazioni di ripristino della DRAM e la diversa organizzazione delle celle. La RAM statica spesso è organizzata in byte, cioè possono essere memorizzati in un circuito integrato 8 bit contemporaneamente, mentre la DRAM solitamente è organizzata in bit cosicché essa ha un certo numero di posizioni, ognuna delle quali memorizza un bit. Ciò significa che un byte di 8 bit è suddiviso in 8 circuiti integrati collegati in parallelo invece che essere interamente contenuto in un solo circuito integrato. Ciò modifica l'analisi dei risultati del test. Una linea bloccata a 0 o ad 1 in una RAM statica potrebbe essere causata sia dal buffer di uscita in quella posizione di bit del circuito integrato della RAM o dal cablaggio su quella linea. Nel caso della DRAM il malfunzionamento di un singolo cirsegnali di temporizzazione e di controllo.

cuito integrato (un problema nel suo decodificatore di ingresso o nei circuiti del buffer di temporizzazione) farebbe risultare guasta la posizione di tutti i bit assegnati ad esso per tutte le locazioni di memoria all'interno della gamma di indirizzo della memoria dinamica. Perciò potrebbe essere possibile determinare anche più velocemente quale circuito è guasto in quanto sarà in generale sbagliato lo stesso bit per tutti gli indirizzi controllati. Naturalmente il problema potrebbe essere situato nel cablaggio del circuito, proprio come poteva avvenire per la ROM o per la RAM statica. Controllare il funzionamento dell'operazione di ripristino della DRAM richiede tecniche differenti che verranno trattate in uno dei prossimi capitoli. L'altro aspetto della prova di una DRAM è che le particolari relazioni tra la temporizzazione e i segnali di controllo (*figura 6.8*) rendono un po' più difficile costruire un dispositivo speciale di prova del tipo mostrato in *figura 6.10*. È molto più conveniente, anche dal punto di vista economico, utilizzare programmi di prova automatici del tipo mostrato a questo scopo in *figura 6.11*. Il programma di prova potrebbe risiedere nello strumento di test basato su microprocessore o potrebbe far parte della sequenza diagnostica che viene avviata all'accensione del sistema. Questo sistema di autodiagnosi verrà trattato più ampiamente nell'ultimo capitolo.

CHE COSA ABBIAMO IMPARATO?

1. I sottosistemi di memoria a semiconduttori sono composti da circuiti di memoria e da decodificatori e porte.
2. Le verifiche del funzionamento di RAM statiche, RAM dinamiche e ROM sono simili.
3. La parte di logica combinatoria della memoria può essere provata utilizzando le tecniche illustrate nei capitoli precedenti, in particolare nel capitolo 4.
4. Verificare il funzionamento dei dispositivi di memoria è un po' più complicato perché di solito si devono controllare molte migliaia di locazioni e si deve trarre da queste prove una sintesi esauriente.
5. È possibile compiere la prova manualmente con un semplice strumento di test; comunque verificare il funzionamento di una memoria composta da molte migliaia di locazioni potrebbe richiedere ore.
6. Un microprocessore e uno speciale programma di prova possono determinare velocemente e facilmente se i sistemi RAM e ROM funzionano correttamente.
7. La prova automatica può fornire un'analisi e risultati dettagliati che aiutano il riparatore ad individuare velocemente il guasto.

Quiz del Capitolo 6

1. Una memoria statica memorizzerà informazioni:
   a. Anche quando la memoria non viene alimentata.
   b. Finché la memoria viene alimentata.
   c. Finché la memoria viene alimentata e ripristinata periodicamente.

2. Una memoria dinamica memorizzerà informazioni:
   a. Anche quando la memoria non viene alimentata.
   b. Finché la memoria viene alimentata.
   c. Finché la memoria viene alimentata e ripristinata periodicamente.

3. Il numero tipico dei bit in una locazione di memoria statica è:
   a. 1
   b. 2
   c. 8
   d. 16

4. Il numero tipico dei bit in una posizione di memoria dinamica è:
   a. 1
   b. 2
   c. 8
   d. 16

5. Una memoria statica generalmente contiene:
   a. Nessun decodificatore.
   b. Decodificatori di riga.
   c. Decodificatori di colonna.
   d. Decodificatori sia di riga che di colonna.

6. Una memoria dinamica generalmente contiene:
   a. Nessun decodificatore.
   b. Decodificatori di riga.
   c. Decodificatori di colonna.
   d. Decodificatori sia di riga che di colonna.

7. Se il processore di *figura 6.3* fornisse 16 bit di indirizzo, quanti circuiti di memoria sarebbero richiesti per ottenere una memoria che utilizzi tutti gli indirizzi possibili?
   a. 4
   b. 8
   c. 16
   d. 32

8. Se, nel caso della domanda precedente, la struttura di *figura 6.3* fosse usata con i bit di indirizzo più significativi inviati ad una porta NAND per generare il segnale di abilitazione $\overline{E}$ per il decodificatore, quale sarebbe l'indirizzo della prima locazione nel circuito di memoria all'estremo superiore?
   a. 0000
   b. 2000
   c. A000
   d. E000
   e. F000

9. Se il segnale $\overline{E}$ di *figura 6.6* fosse generato quando le linee di indirizzo più significative contengono il codice 10_2, quale sarebbe l'indirizzo dell'ultima locazione della memoria mostrata?
   a. 0000
   b. 4FFF
   c. 8000
   d. 8FFF
   e. BFFF

10. Se il circuito di prova di memoria mostrato in *figura 6.10* è in grado di controllare una locazione al secondo, quante ore occorreranno per verificare il funzionamento della memoria mostrata in *figura 6.3*?
    a. 0,4
    b. 2,22
    c. 133,3
    d. 8000

11. Se per eseguire ogni istruzione di *figura 6.11* occorrono 2 microsecondi, quanti secondi occorrerebbero approssimativamente per provare una memoria non guasta del tipo mostrato in *figura 6.3*?
    a. 0,006
    b. 0,06
    c. 0,6
    d. 6
    e. 60

12. Se l'unità di controllo del ripristino di una memoria dinamica mostrata in *figura 6.9* venisse utilizzata con un circuito di memoria da 64K, quale sarebbe in microsecondi l'intervallo di ripristino?
    a. 2
    b. 4
    c. 8
    d. 16
    e. 32

13. Nel caso della domanda precedente, quale sarebbe il numero di bit di indirizzo di riga al di fuori del contatore e del multiplexer?
    a. 7
    b. 8
    c. 9

d. 10
e. 16

14. Se nel sistema di *figura 6.3* nessuna delle memorie risponde ad un accesso alla memoria da parte del processore, il guasto più probabile sarebbe:
   a. La linea 0 di uscita del decodificatore guasta.
   b. I dispositivi di memoria guasti.
   c. Il decodificatore guasto.
   d. La porta di abilitazione guasta.
   e. O la risposta c o la risposta d.

15. Se nel sistema di memoria di *figura 6.6* il bit D_3 è sempre 0, il difetto più probabile è:
   a. Il decodificatore guasto.
   b. Il generatore del segnale $\overline{RAS}$ guasto.
   c. Il circuito di memoria nella posizione D_3 guasto.
   d. Il generatore del segnale $\overline{CAS}$ guasto.

16. Se una memoria è formata da 4 righe di circuiti del tipo mostrato in *figura 6.6* con un decodificatore 2 a 4 e si osserva che la memoria è guasta nella gamma di indirizzi compresa tra 0000 e 3FFF, il guasto più probabile è:
   a. Un guasto sulla linea di uscita 0 del decodificatore.
   b. Il decodificatore guasto.
   c. Il generatore del segnale $\overline{CAS}$ guasto.
   d. La riga dei circuiti di memoria nella prima riga della memoria guasta.

17. Per una memoria del tipo di quella della domanda 16, se il guasto fosse che D_3 è errato quando accede alla gamma di indirizzi compresi fra 8000 e BFFF, esso sarà probabilmente dato:
   a. Dal decodificatore guasto.
   b. Da un circuito aperto su una delle linee di uscita del decodificatore.
   c. Dal generatore del segnale $\overline{RAS}$ guasto.
   d. Da un circuito di memoria difettoso nella terza riga della colonna D_3
   e. Da un circuito di memoria difettoso nella seconda riga della colonna D_3

18. Se la memoria descritta nel problema 16 è guasta, il problema più probabile è:
   a. Il decodificatore guasto.
   b. Un circuito aperto sulle linee di uscita del decodificatore.
   c. Il generatore del segnale $\overline{RAS}$ guasto.
   d. Un circuito di memoria difettoso nella terza riga della colonna D_3
   e. Un circuito di memoria difettoso nella seconda riga della colonna D_3

19. Nel programma di prova mostrato in *figura 6.11* la strategia è quella di:
   a. Inviare alternativamente degli 1 e degli 0 a ciascuna locazione di memoria e leggere queste locazioni per controllare se i valori sono stati memorizzati correttamente; quindi ripetere il test inviando una configurazione in cui gli 1 e gli 0 siano stati scambiati fra loro.
   b. Inviare tutti 0 a ciascuna locazione di memoria e verificare la capacità di memorizzare degli 0; quindi ripetere il test inviando una configurazione di tutti 1.
   c. Non fare nessuna delle due cose descritte precedentemente.

20. Se la strategia del programma di prova di *figura 6.11* viene trasformata dalla 19a alla 19b o viceversa, quale istruzione dovrebbe essere cambiata?
   a. LI 3,5555
   b. INV 3
   c. XOR 3,5
   d. CLR 2

ASPETTI DI INGRESSO/ USCITA (I/O)

SCOPO DEL CAPITOLO

Nell'ultimo capitolo sono stati trattati il funzionamento e i problemi principali delle memorie di un microcomputer. Gli scopi principali del sottosistema di ingresso e uscita sono stati introdotti nella prima parte del capitolo 5. In questo capitolo si considereranno invece i tipi di configurazioni dei sistemi di ingresso e uscita e le tecniche di interfacciamento che si incontrano nei progetti dei microcomputer insieme ad alcuni tipici guasti e alle tecniche per individuare questi ultimi in tali sistemi.

TIPI DI COMPONENTI DI INGRESSO/USCITA E RELATIVI PROBLEMI

In un microcomputer i circuiti di ingresso e uscita sono controllati dal microprocessore tramite le linee del bus indirizzi e del bus di controllo. L'informazione viene trasferita tra il processore e i dispositivi di ingresso e di uscita lungo le linee del bus dati. In questa parte del capitolo considereremo i tipi di trasferimento dell'informazione ed i circuiti che accoppiano il processore e i bus.

Si possono usare diversi metodi per controllare i dispositivi di ingresso e uscita in un microcomputer. Generalmente questi dispositivi sono formati da registri e da buffer a tre stati, sebbene siano usati anche decodificatori, codificatori e multiplexer. I tre metodi comunemente usati per controllare l'accesso a questi dispositivi di ingresso e uscita sono: 1) I/O (input/output, cioè ingresso/uscita) a mappa di memoria; 2) trasferimento multibit a mappa di I/O; 3) trasferimento I/O a mappa di bit. Non è che ogni processore usi tutti questi metodi, ma tutti i processori permettono l'uso del metodo a mappa di memoria. Quando quest'ultimo metodo viene utilizzato, al dispositivo I/O viene assegnato un indirizzo di memoria come se esso fosse una locazione nella memoria. Quindi il dispositivo I/O viene collegato con i segnali del bus indirizzi e del bus di controllo proprio come se fosse un dispositivo di memoria. Si ricordi che queste cose sono state trattate nel capitolo precedente.

I tre metodi di accesso ai dispositivi di ingresso/uscita più comunemente usati nei sistemi di microcomputer sono: la memoria mappata, i trasferimenti multibit a mappa di I/O e i trasferimenti di I/O a mappa di bit.

Alcuni processori lavoreranno anche con il metodo a mappa di I/O. Quando viene usato questo metodo i registri I/O e i buffer a tre stati hanno un indirizzo che è valido solo come indirizzo I/O. Essi non risponderanno ad un indirizzo di memoria. I trasferimenti da e per i dispositivi I/O si verificano sul bus dati del sistema; perciò sono trasferimenti sia ad 8 bit in parallelo che a 16 bit in parallelo, a seconda del microprocessore. La maggior parte dei microprocessori a 16 bit (che hanno un bus dati a 16 bit) offrono l'alternativa dei trasferimenti a 8 bit o a 16 bit da e verso i registri e da e verso i buffer a tre stati.

Solo pochi processori, inoltre, effettuano i trasferimenti da e verso i singoli bit indirizzabili nel sistema lungo una serie di linee di uscita e di ingresso al singolo bit. Tutti questi diversi tipi di I/O verranno considerati in questa parte del capitolo insieme ai problemi che possono verificarsi nel funzionamento di tali circuiti.

Uscita a mappa di bit

Il primo esempio che verrà considerato è il circuito di uscita a mappa di bit mostrato in *figura 7.1*. In questo caso il microprocessore è il TMS9980, cioè un processore a 16 bit con un bus dati a 8 bit. Questo processore ha la capacità di indirizzare 2048 bit individuali di ingresso e uscita. Il trasferimento di bit si verifica sulle linee CRUIN (linea di ingresso dei bit) e CRUOUT (linea di uscita dei bit). Il processore inoltre emette un segnale CRUCLK per indicare quando un dato valido sta apparendo sulla linea CRUOUT.

Nell'esempio dell'uscita a mappa di bit, un flip-flop di un latch ottale indirizzabile ha il suo segnale di latch attivato dall'uscita di un decodificatore 3 a 8 per catturare i dati inviati a tutti i flip-flop.

Il dispositivo di selezione dell'uscita di *figura 7.1* è un latch ottale indirizzabile. Le linee di indirizzo corrispondenti ai tre bit meno significativi dell'indirizzo sono collegate con le tre linee del codice di selezione per scegliere la linea di uscita. Il segnale CRUCLK viene utilizzato per abilitare il circuito a memorizzare il bit presente su CRUOUT sulla linea di uscita selezionata. Perciò quando CRUCLK compie una transizione da un livello alto a un livello basso, il flip-flop all'interno del dispositivo che riceve lo 0 o l' 1 su CRUOUT verrà determinato dal codice di selezione a 3 bit. Il bit può essere memorizzato nel flip-flop selezionato solo quando il segnale di abilitazione del dispositivo si trova nel suo stato attivo.

Come mostrato in *figura 7.1* il latch ottale indirizzabile può essere pensato come un decodificatore 3 ad 8 le cui uscite vengono inviate agli ingressi di latch (clock) di 8 flip-flop di tipo D. La linea di latch viene selezionata dal codice di ingresso del decodificatore 3 a 8 e abilitata dal segnale presente sul piedino di abilitazione del decodificatore. Il segnale CRUOUT viene inviato all'ingresso D di tutti ed 8 i flip-flop, ma viene memorizzato solo nel flip-flop che riceve l'impulso di latch. L'impulso di latch sulla linea selezionata è una copia invertita del segnale CRUCLK.

In *figura 7.1* si noti che le 8 linee di indirizzo più significative vengono inviate ad una porta NAND insieme al segnale CRUCLK. Perciò quando tutti questi segnali sono alti l'uscita della porta NAND andrà ad un livello basso abilitando il decodificatore e l'intero latch ottale indirizzabile. Perciò il latch ottale verrà abilitato solo quando il codice di indirizzo è:

111 1111 1XXX

dove XXX è il codice di selezione a 3 bit del flip-flop. L'indirizzo completo del flip-flop (FF1) in alto nella *figura 7.1* sarebbe 7F8 (esadecimale) perché XXX = 000. L'indirizzo completo del flip-flop (FF8) in basso in *figura 7.1* sarebbe 7FF (esadecimale) perché XXX = 111. Questo è il metodo generale con cui si determina l'indirizzo per ogni specifico bit nel sottosistema I/O a mappa di bit.

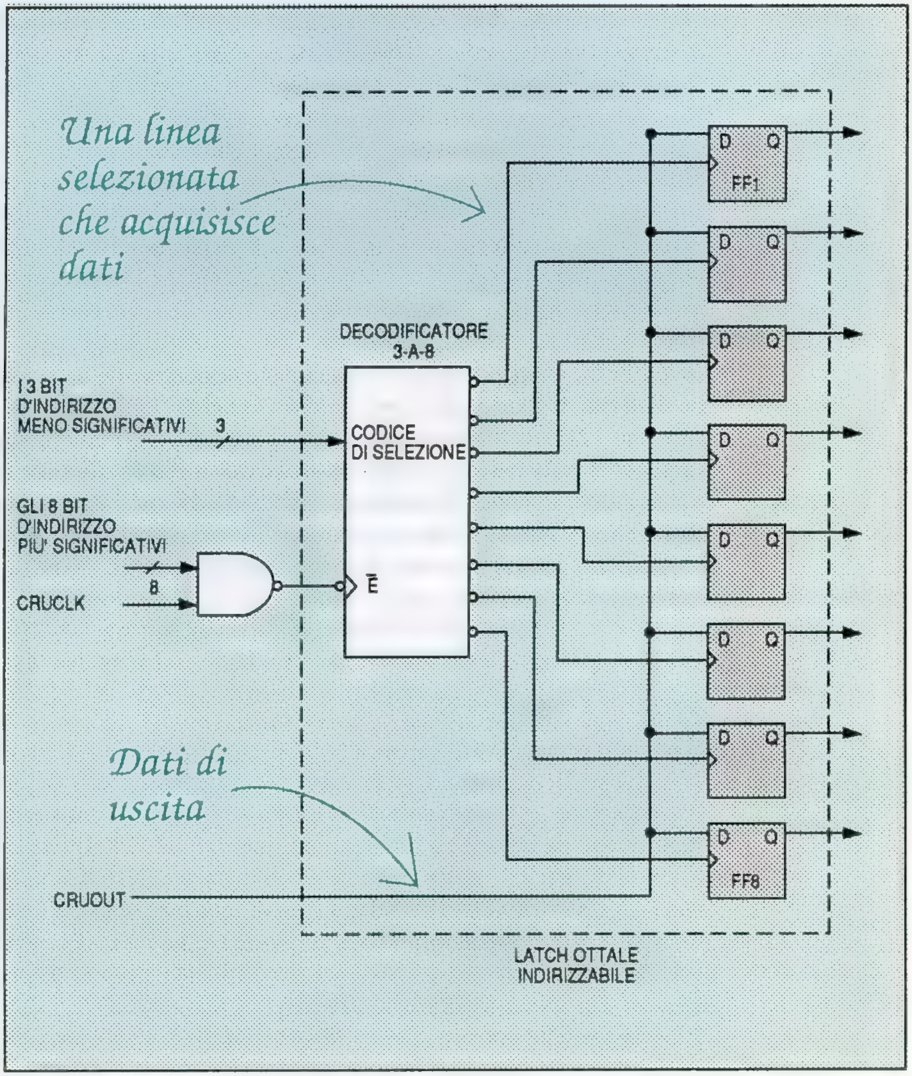
> N.B.: Il termine esadecimale viene solitamente abbreviato con HEX, (H), o H. Nella restante parte di questo libro verrà utilizzato il simbolo H per la parola esadecimale (scriveremo, per esempio, 7FFH).

Per inviare uno 0 a questi bit utilizzando le istruzioni del microprocessore TMS9980, si dovrebbe programmare l'istruzione:

SBZ n

così da portare il bit indicato con n a 0. Il bit che riceve lo 0 viene identificato dal codice di indirizzo che si ottiene aggiungendo il valore di n letto dall'istruzione all'indirizzo di base contenuto nel registro 12 (R12) del processore. Perciò se R12 contiene l'indirizzo di base 7F0H e viene eseguita l'istruzione SBZ 8, l'indirizzo del bit sarà 7F0H + 8 cioè 7F8H. Il bit nell'indirizzo 7F8H diventerà allora 0. In altre parole

Figura 7.1. Circuito d'uscita a mappa di bit


Una linea selezionata che acquisisce dati
DECODIFICATORE 3-A-8
I 3 BIT D'INDIRIZZO MENO SIGNIFICATIVI
3
CODICE DI SELEZIONE
GLI 8 BIT D'INDIRIZZO PIU' SIGNIFICATIVI
8
CRUCLK
Ē
D Q
FF1
FF8
Dati di uscita
CRUOUT
LATCH OTTALE INDIRIZZABILE

il flip-flop in alto nella *figura 7.1* (FF1) riceverà uno 0 sulla linea CRUOUT. Analogamente l'istruzione:

SBO n

porterebbe il bit indirizzato ad 1. Perciò se R12 contiene l'indirizzo di base 7F0H e viene eseguita l'istruzione SBO 15, il bit all'indirizzo 7FFH (7F0 + F) verrà portato ad 1. In altre parole il flip-flop in basso nella *figura 7.1* (FF8) verrà portato ad 1.

Ricerca ed eliminazione dei guasti

I guasti più comuni dei flip-flop sono quelli in cui uno o più flip-flop sono bloccati ad uno dei due livelli. Se tutti i flip-flop sbagliano nella risposta, la causa è qualcosa di comune a tutti.

Dal punto di vista della ricerca dei guasti il funzionamento del circuito di *figura 7.1* è relativamente semplice. Il problema che si presenta più comunemente è che i flip-flop non ricevano né 0 né 1 ma siano bloccati al livello 0 o al livello 1. Se tutti i flip-flop non rispondono alle istruzioni di uscita allora il guasto non riguarda i flip-flop, ma dovrà trovarsi in una parte del circuito che è comune a tutti flip-flop. Il problema dovrebbe dunque essere o sulla linea di ingresso di abilitazione o sulla linea di ingresso D. Un'apertura su una delle due linee farebbe sì che il circuito risponda in modo errato ad un'istruzione di uscita 0 o 1. Il guasto relativo all'abilitazione potrebbe essere dovuto ad un guasto nella porta NAND la quale decodifica le linee di indirizzo più significative e fa passare il segnale CRUCLK attraverso il piedino di abilitazione.

Se tutti i flip-flop, eccetto uno, funzionano normalmente in risposta ad un'istruzione di uscita, il guasto potrebbe trovarsi: (1) nel decodificatore che genera il latch al flip-flop, (2) sulla linea che collega il segnale CRUOUT con l'ingresso del flip-flop, oppure (3) sulla linea che collega l'uscita del decodificatore con l'ingresso di latch del flip-flop. In un latch ottale indirizzabile, una qualsiasi di queste disfunzioni indicherebbe il guasto di un componente all'interno del circuito integrato. In ogni caso, il guasto può essere trovato abbastanza facilmente con delle semplici tecniche di misurazione.

Ingresso a mappa di bit

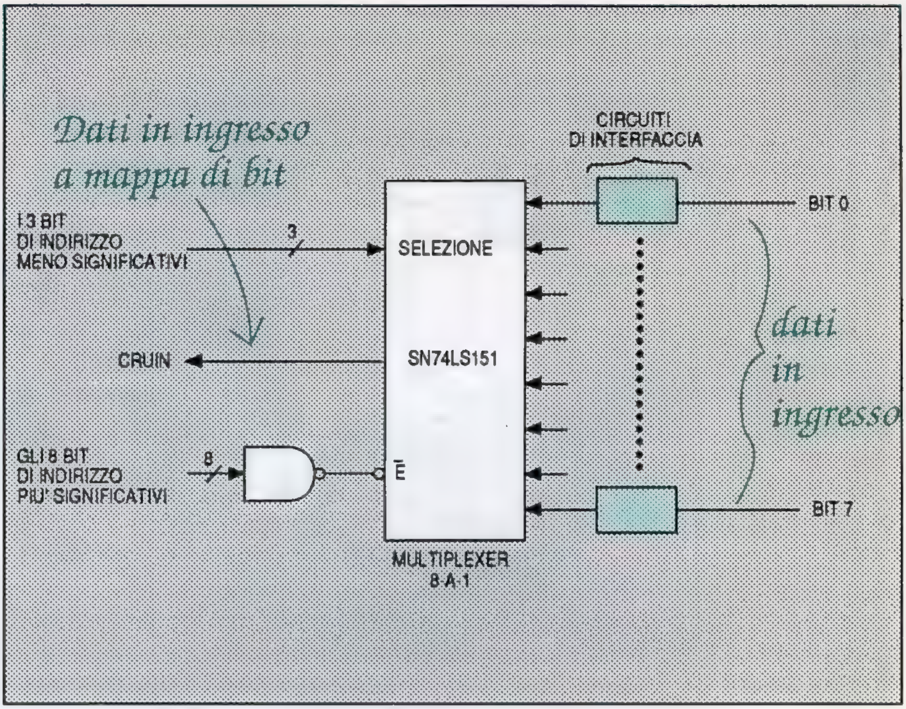
Nell'esempio dell'ingresso a mappa di bit un decodificatore multiplexer 8 ad 1 seleziona l'unica fra le 8 linee d'ingresso che deve essere inviata alla linea CRUIN.

Il circuito di ingresso a mappa di bit di *figura 7.2* è molto simile al circuito di uscita di *figura 7.1*. Il dispositivo di selezione di ingresso è un multiplexer del tipo 8 linee ad 1 linea. In questo esempio la porta abilitante è ancora una porta NAND e il multiplexer ha la stessa gamma di indirizzi (7F8H-7FFH) che aveva il circuito di uscita di *figura 7.1*. Non si verifica alcun ingresso CRUCLK alla porta NAND finché l'ingresso è abilitato solo dagli 8 bit di indirizzo più significativi. L'ingresso al microprocessore viene controllato al registro di ingresso del microprocessore. I tre bit di indirizzo meno significativi vengono utilizzati per selezionare quale degli 8 ingressi viene passato sulla linea CRUIN.

L'istruzione che immette lo 0 o l'1 su una delle 8 linee di ingresso è:

TB n

Figura 7.2. Circuito d'ingresso a mappa di bit


Dati in ingresso a mappa di bit
CIRCUITI DI INTERFACCIA
I 3 BIT DI INDIRIZZO MENO SIGNIFICATIVI
3
SELEZIONE
BIT 0
CRUIN
SN74LS151
dati in ingresso
GLI 8 BIT DI INDIRIZZO PIU' SIGNIFICATIVI
8
Ē
BIT 7
MULTIPLEXER 8-A-1

dove, ancora una volta, n viene aggiunto all'indirizzo di base del registro 12 del processore per ottenere l'indirizzo del bit selezionato. Per esempio, se l'indirizzo di base nel registro 12 è 7F8H e si deve inserire il bit 7, sarà necessaria l'istruzione TB 7 per ottenere l'indirizzo 7FFH (7F8 + 7 = 7FF).

I problemi che possono verificarsi in questo tipo di circuito sono limitati ai guasti del multiplexer, ai guasti della porta NAND, oppure ad un corto circuito o ad un circuito aperto su una delle linee del circuito. Se nessuno dei bit può essere immesso sulla linea CRUIN il guasto è situato nei collegamenti e nella porta di abilitazione. Se invece non si può inviare l'ingresso a CRUIN su di una sola linea, il guasto deve trovarsi nel collegamento di quella linea con il multiplexer o nel multiplexer stesso. Il malfunzionamento di due ingressi dovrebbe essere invece dovuto ad un guasto in una delle linee di selezione che provoca il blocco a 0 o ad 1 del codice di selezione.

I/O in parallelo

Il metodo I/O in parallelo (trasferimento multibit a mappa di I/O) differisce dal metodo a mappa di bit perché il trasferimento I/O in parallelo si verifica sul bus dati del microprocessore come un trasferimento di memoria. La differenza sta nel fatto che l'I/O in parallelo risponde ad un segnale di abilitazione I/O, mentre l'I/O a mappa di memoria risponde ad un segnale di abilitazione di memoria.

Figura 7.3. Circuito d'ingresso a mappa di memoria

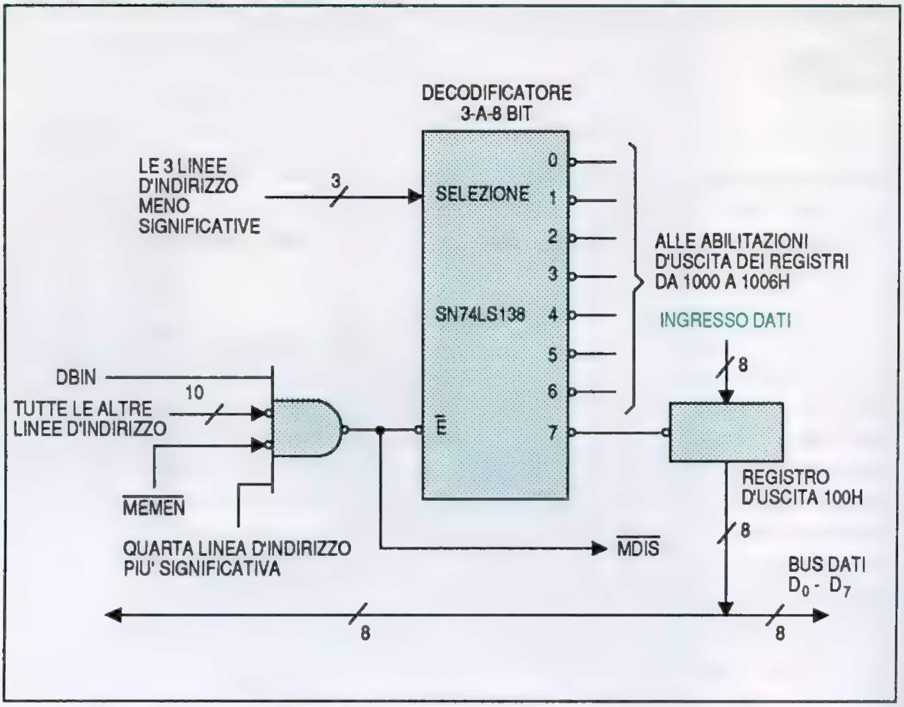
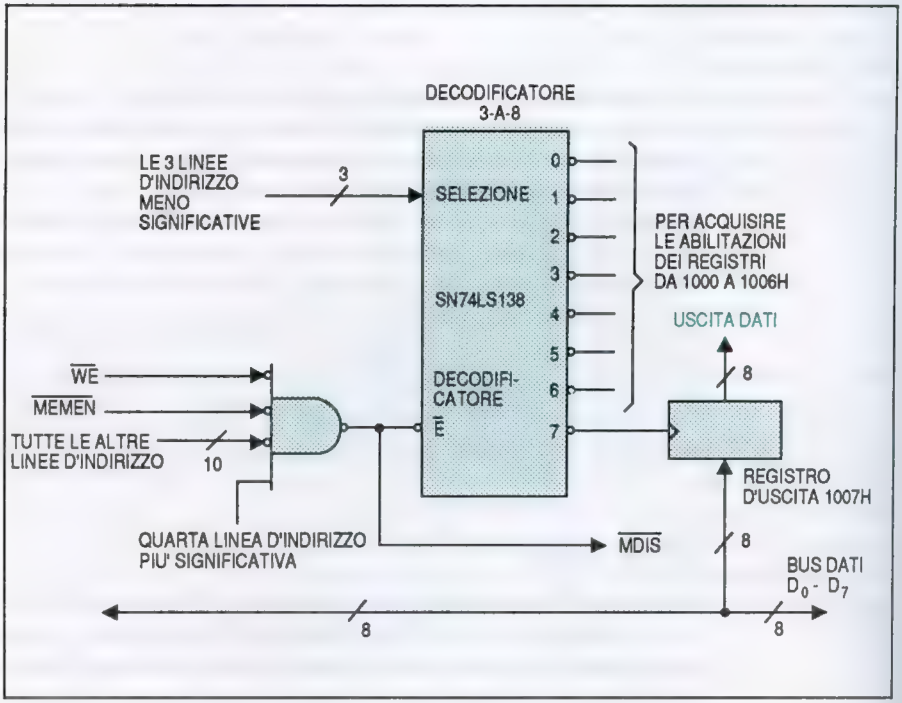

DECODIFICATORE
3-A-8 BIT
LE 3 LINEE
D'INDIRIZZO
MENO
SIGNIFICATIVE
3
SELEZIONE
0
1
2
3
4
5
6
7
SN74LS138
$\overline{E}$
ALLE ABILITAZIONI
D'USCITA DEI REGISTRI
DA 1000 A 1006H
INGRESSO DATI
8
DBIN
10
TUTTE LE ALTRE
LINEE D'INDIRIZZO
$\overline{MEMEN}$
QUARTA LINEA D'INDIRIZZO
PIU' SIGNIFICATIVA
$\overline{MDIS}$
REGISTRO
D'USCITA 100H
8
BUS DATI
D_0 - D_7
8
8

Figura 7.4. Circuito d'uscita a mappa di memoria


DECODIFICATORE
3-A-8
LE 3 LINEE
D'INDIRIZZO
MENO
SIGNIFICATIVE
3
SELEZIONE
0
1
2
3
4
5
6
7
SN74LS138
DECODIFI-
CATORE
$\overline{E}$
PER ACQUISIRE
LE ABILITAZIONI
DEI REGISTRI
DA 1000 A 1006H
USCITA DATI
8
$\overline{WE}$
$\overline{MEMEN}$
TUTTE LE ALTRE
LINEE D'INDIRIZZO
10
QUARTA LINEA D'INDIRIZZO
PIU' SIGNIFICATIVA
$\overline{MDIS}$
REGISTRO
D'USCITA 1007H
8
BUS DATI
D_0 - D_7
8
8

I/O a mappa di memoria

Nel sistema I/O a mappa di memoria il registro d'ingresso I/O e il registro d'uscita I/O sono indirizzati proprio come una memoria. Un segnale di controllo determina se l'indirizzo è per un dispositivo di I/O o per la memoria.

La struttura fondamentale di un registro di ingresso a mappa di memoria è mostrata in *figura 7.3* per un processore di tipo TMS9980. La *figura 7.4* mostra invece la struttura fondamentale di un registro di uscita a mappa di memoria. Sia al registro di ingresso che a quello di uscita sono assegnati gli indirizzi di memoria da 1000H a 1007H. Potrebbe anche esserci un circuito di memoria con indirizzi da 1000H a 1007H, ma il circuito di memoria dovrebbe venire disabilitato quando questi indirizzi vengono usati in un'operazione di I/O. Questo è infatti lo scopo della linea di disabilitazione di memoria ($\overline{\text{MDIS}}$) di *figura 7.3* e di *figura 7.4*.

Nel circuito di ingresso di *figura 7.3*, il decodificatore risponde con un livello basso su una delle sue linee di uscita per abilitare un registro di ingresso quando si verificano tutte le seguenti condizioni: (1) è indirizzato un indirizzo di memoria compreso tra 1000H e 1007H; (2) il segnale di abilitazione di memoria (MEMEN) è attivo; (3) il segnale di abilitazione al trasferimento in ingresso DBIN è attivo; (4) le linee di indirizzo più significative sono attive. Perciò il registro di ingresso invierà i suoi dati al bus dati perché siano trasferiti ad uno dei registri del processore o ad una delle locazioni di memoria quando viene eseguita un'istruzione di ingresso.

In *figura 7.4*, il decodificatore che controlla i registri di uscita viene abilitato sotto queste stesse condizioni, eccetto che al posto del segnale DBIN viene utilizzato il segnale di abilitazione di scrittura $\overline{\text{WE}}$. Il processore invierà i dati al bus dati e il registro di uscita selezionato memorizzerà questi dati quando verrà eseguita un'istruzione di uscita.

Per esempio per immettere i dati provenienti dal registro esterno a 1002H alla locazione di memoria 2000H sarà necessaria l'esecuzione dell'istruzione:

MOV @ > 1002, @ > 2000

Per inviare invece i dati provenienti dal registro 2 del processore al registro di uscita situato alla locazione 1007H sarà necessario eseguire l'istruzione:

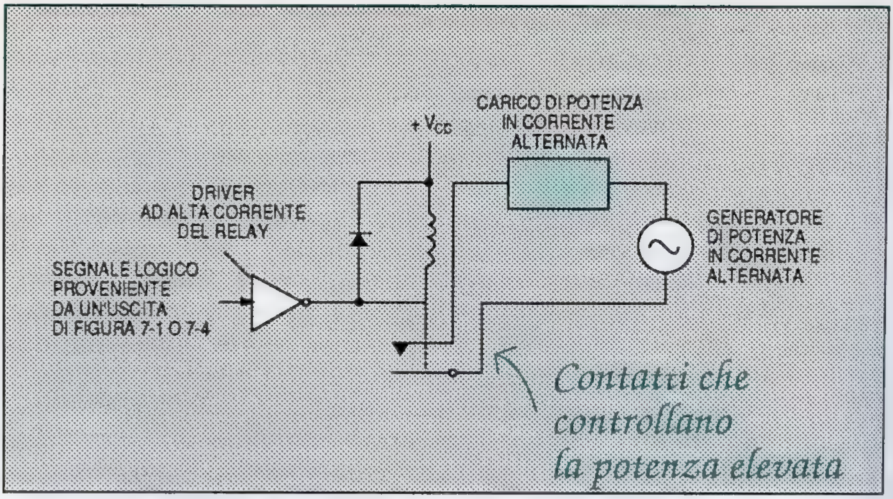
MOV 2, @ > 1007

In questo modo è relativamente semplice mettere in relazione l'istruzione con l'operazione di trasferimento di ingresso o di uscita.

I circuiti d'interfaccia convertono i segnali digitali in livelli di segnale utilizzati da altri sistemi. I circuiti di controllo convertono le uscite in azioni fisiche.

Indipendentemente da quale metodo di ingresso/uscita venga utilizzato, questo permette al microprocessore di ricevere o di inviare un singolo segnale o una sequenza di segnali digitali da e verso un altro sistema o dispositivo di controllo. Perché questi segnali siano utili per il controllo o le comunicazioni, i livelli di tensione dei segnali digitali devono essere cambiati in altri livelli di tensione. I circuiti compresi fra i registri di ingresso/uscita e gli altri sistemi che eseguono questo cambiamento di tensione o che modificano comunque i segnali vengono chiamati circuiti di interfaccia. I circuiti che utilizzano i segnali di uscita per provocare un funzionamento fisico (come per esempio l'accensione di un motore) vengono chiamati circuiti di controllo. Sia i circuiti di interfaccia che i circuiti di controllo possono essere fonte di guasti nel sistema; perciò questi circuiti e il loro funzionamento devono essere compresi.

Figura 7.5. Circuito di controllo di potenza in AC



Circuito di controllo

Un circuito di controllo piuttosto comune è un relay alimentato da un 1 logico in uscita dal computer, i cui contatti controllano un circuito ad alta corrente o ad alta tensione.

La *figura 7.5* illustra un tipo di circuito di uscita che si incontra spesso nei sistemi che devono controllare carichi ad alta potenza come le caldaie o i motori. Il driver dei relay ad alta corrente risponde a livelli di tensione digitali presenti al suo ingresso e fornisce un grande flusso di corrente nel circuito di uscita quando è presente un 1 all'ingresso. Questa corrente può essere utilizzata per alimentare la bobina di un relay da 24 fino a 28 V in corrente continua al fine di provocare la chiusura dei contatti del relay. Ciò attiva la potenza in corrente alternata nella caldaia, nel motore o in ogni altro carico sotto controllo in modo che venga fornita l'azione desiderata. Il diodo viene collegato in parallelo con la bobina, in modo da evitare che il driver del relay venga danneggiato dalle alte tensioni che si producono quando viene interrotta la corrente della bobina. La polarità di questa tensione è opposta a quella della tensione di alimentazione: perciò il diodo conduce e dissipa l'energia.

Scegliendo un relay con valori sufficienti di tensione e corrente sui suoi contatti, quasi tutti i livelli di potenza possono essere controllati dal microcomputer. Per controllare carichi ad alta potenza vengono utilizzati anche circuiti simili che utilizzano al posto del relay un SCR o un TRIAC con un trasformatore di impulsi o un accoppiatore ottico per isolare il computer. Questi sono un po' più complicati del circuito di *figura 7.5*, ma hanno la stessa funzione e modalità di guasto simili.

Ricerca ed eliminazione dei guasti nel circuito di controllo

Nel circuito di un relay le più probabili cause di guasto sono una bobina difettosa, i contatti o il driver.

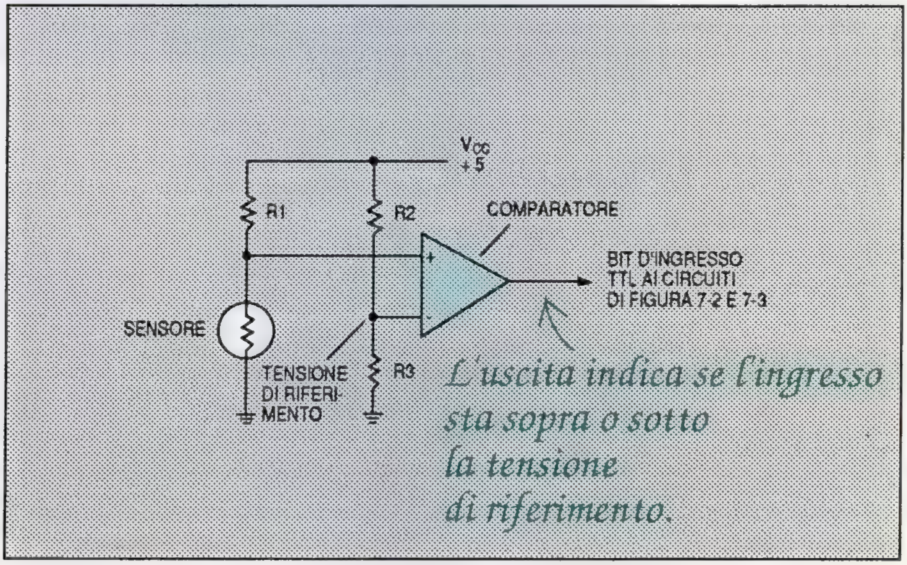
In un circuito del tipo mostrato in *figura 7.5* la più probabile causa di guasto è un'apertura nell'avvolgimento del relay, o contatti del relay guasti, o un driver del relay malfunzionante. Questi guasti potranno essere controllati sconnettendo l'ingresso del driver del relay dal microcomputer e applicando una tensione di livello 0 all'ingresso del driver. Il passaggio di corrente nella bobina dovrebbe essere 0 e i contatti del relay dovrebbero indicare un circuito aperto (nessun passaggio di corrente nel circuito di potenza). All'uscita del driver del relay dovrebbe esserci una tensione di V_{cc} volt.

Se l'uscita del driver del relay è a 0 volt e il relay è chiuso, c'è un corto circuito nel circuito di uscita del driver e quindi il driver dovrebbe essere sostituito. Il guasto del driver potrebbe essere stato provocato dall'apertura del diodo collegato in parallelo alla bobina del relay; perciò il diodo dovrebbe essere scollegato e controllato con un ohmetro prima di sostituire il driver. Il diodo, col terminale negativo collegato col catodo (barra) e quello positivo collegato con l'anodo (punta della freccia), dovrebbe avere una resistenza bassa. (Si ricordi che alcuni ohmetri hanno come positivo la boccola nera). Quando vengono invertiti i collegamenti dovrebbe essere indicata una resistenza alta (almeno 100 volte la resistenza bassa). Se ciò non accade il diodo è difettoso.

Se l'uscita del driver è V_{cc} quando è applicato uno 0 al suo ingresso, allora dovrebbe essere applicato un livello 1 all'ingresso del driver stesso. Questo dovrebbe provocare il passaggio della corrente attraverso la bobina del relay, una tensione all'uscita del driver di circa 0 volt e la chiusura dei contatti del relay. Se la tensione all'uscita del driver è V_{cc}, la bobina del relay potrebbe essere aperta o ci potrebbe essere un'apertura nel circuito del driver, nel qual caso il driver sarebbe guasto e dovrebbe essere sostituito. Comunque prima di sostituire la bobina o il driver si controlli il diodo. Esso deve funzionare correttamente perché possano essere prevenuti i danni al driver. Se il driver funziona correttamente, ma i contatti del relay non sono chiusi, allora c'è probabilmente un guasto nella bobina dl relay o nei suoi contatti oppure il diodo ha subito un corto circuito. Dopo che la parte difettosa è stata sostituita, si controlli nuovamente il circuito per verificare che funzioni correttamente prima di ricollegare l'ingresso del driver con l'uscita del microcomputer.

Circuito sensore

La *figura 7.6* illustra un tipo di circuito di ingresso per convertire segnali analogici



L'uscita indica se l'ingresso sta sopra o sotto la tensione di riferimento.

Figura 7.6. Circuito sensore rivelatore di soglia

in segnali digitali. Un segnale non elettrico, come per esempio la temperatura, viene convertito in tensione elettrica attraverso l'uso di un sensore e di un circuito partitore di tensione resistivo. Poiché il passaggio di corrente attraverso il circuito di rilevamento viene controllato dalla variabile che si sta rilevando (temperatura o altra quantità), la tensione all'ingresso + dell'amplificatore operazionale è controllata dalla variabile. Perciò l'uscita dell'amplificatore è, in qualche modo, funzione della variabile rilevata.

Un comune circuito sensore utilizza un amplificatore operazionale come comparatore per individuare il livello di una tensione analogica in ingresso confrontato con un livello di soglia.

In *figura 7.6* l'amplificatore operazionale è effettivamente usato come comparatore. Se la tensione + è maggiore della tensione -, l'uscita del dispositivo è un livello di 1 logico che viene letto dal microprocessore attraverso uno dei suoi ingressi. Analogamente se il livello di tensione dell'ingresso + è minore di quello dell'ingresso -, l'uscita del comparatore sarà al livello di tensione 0. Perciò il circuito di figura 7.6 dice al microprocessore quando la variabile rilevata sta sopra o sotto il livello di riferimento. Questo è il modo più semplice per convertire un segnale analogico in un segnale digitale.

Ricerca ed eliminazione dei guasti nel circuito sensore

Il circuito sensore deve essere controllato misurando le tensioni, stabilendo certe condizioni logiche d'ingresso e cercando se esistono corto circuiti con un ohmetro.

I guasti che possono verificarsi più facilmente nel circuito di *figura 7.6* sono: un sensore guasto; un amplificatore guasto; un'apertura o un corto nei resistori. Questi possono essere individuati tramite semplici tecniche di isolamento e di verifica del funzionamento di un componente e attraverso misurazioni di resistenza. Le tensioni + e - possono essere misurate col sensore collegato. Se una delle due è di 0 volt, c'è un corto a massa nel punto in cui si sta eseguendo il test o un'apertura tra questo punto e la tensione di alimentazione V_{cc} Se invece una delle due è di V_{cc} volt, c'è un corto circuito in R1 o in R2, oppure c'è un circuito aperto nel sensore o in R3.

Il funzionamento del comparatore può essere controllato mettendo a massa il terminale - e una tensione positiva sul terminale + e verificando quindi se l'uscita è al livello 1 come dovrebbe verificarsi per i componenti TTL. Analogamente ponendo uno 0 sul terminale + e una tensione positiva di riferimento sul terminale - si dovrebbe avere in uscita uno 0 logico TTL.

TERMINALI DI I/O

I circuiti fondamentali di ingresso e di uscita considerati nelle *figure* dalla *7.1* alla *7.6* vengono utilizzati per rilevare i segnali in ingresso e controllare le azioni in uscita all'interno di un computer, ma non sono del tipo normalmente usato per comunicare con un computer. Il più comune dispositivo I/O per comunicare con un computer è il terminale del computer. Questo normalmente è formato da una tastiera simile a quella di una macchina per scrivere e da un CRT (tubo a raggi catodici) che presenta uno schermo simile a quello di un televisore. (Alcuni terminali utilizzano invece una combinazione tastiera e stampante così che tutte le informazioni vengono stampate su dei fogli di carta). L'informazione, sotto forma di caratteri e di numeri (cioè sotto forma alfanumerica), appare sullo schermo CRT mentre i comandi e le informazioni del testo vengono inseriti nel computer tramite la tastiera. Solitamente il

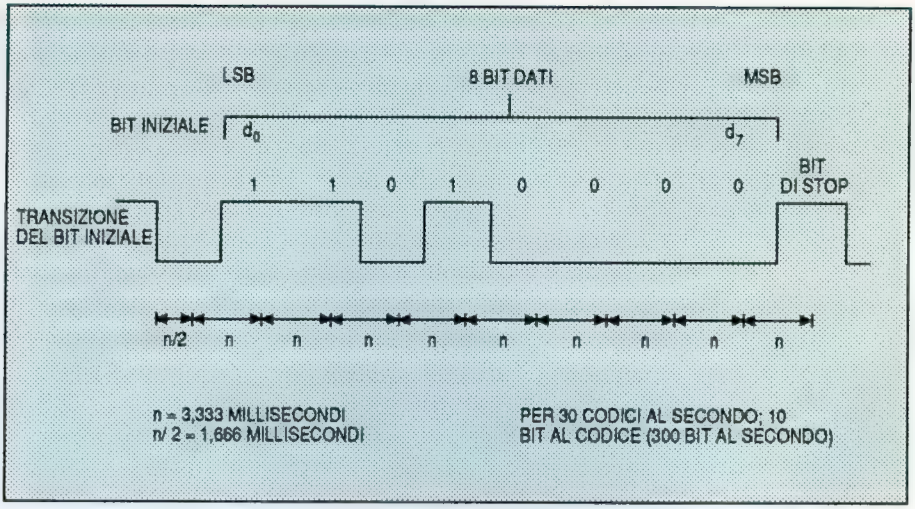

LSB
8 BIT DATI
MSB
BIT INIZIALE
d_0
d_7
1
1
0
1
0
0
0
0
BIT DI STOP
TRANSIZIONE DEL BIT INIZIALE
n/2
n
n
n
n
n
n
n
n
n
n = 3,333 MILLISECONDI
n/ 2 = 1,666 MILLISECONDI
PER 30 CODICI AL SECONDO; 10 BIT AL CODICE (300 BIT AL SECONDO)

Figura 7.7. Trasmissione seriale asincrona dei dati

collegamento tra il computer e il terminale avviene attraverso un collegamento di dati in serie che utilizza un protocollo standard di livelli di segnali logici e di temporizzazione noto come interfaccia RS232. Qui di seguito verrà trattato il funzionamento di un terminale CRT con un'interfaccia RS232 e una tastiera.

Trasferimento seriale dei dati

Nel trasferimento seriale dei dati, il segnale binario viene trasmesso su di un'unica linea e la variazione del segnale nel tempo fornisce la trasmissione del codice binario di lunghezza desiderata. In *figura 7.7* è mostrata una configurazione seriale di dati comunemente usata per la trasmissione di dati asincrona.

Il trasferimento seriale di dati presenta bit di dati frapposti fra il bit iniziale e il bit di stop, riprodotti per ciascun carattere in sequenza.

L'andamento del segnale comincia con un cambiamento della tensione della linea: questa infatti passa dal livello logico 1 al livello logico 0. Il ricevitore rileva questo cambiamento come la transizione del bit iniziale. Dopo un certo tempo, la tensione della linea viene controllata nuovamente per verificare che la tensione è ancora 0. Se è così, il ricevitore ha individuato un bit iniziale valido generato dal trasmettitore. Da questo punto il trasmettitore e il ricevitore concordano che ogni n secondi ci sarà un bit di dato stabile o di 0 o di 1, dove n è il reciproco della velocità di trasmissione espressa in bit al secondo. Nell'esempio di *figura 7.7* la velocità di trasmissione è di 300 bit al secondo (solitamente detti 300 baud) cosicché n è pari a 3,333 millisecondi (1/300).

Il bit meno significativo (LSB) del codice viene inviato per primo, mentre quello più significativo (MSB) per ultimo cosicché il codice ad 8 bit che viene trasmesso nell'esempio di *figura 7.7* è 0000 1011 binario o 0BH. (Si ricordi che il bit LSB viene normalmente scritto come il bit più a destra). Il codice ad 8 bit deve essere seguito da n o più intervalli durante i quali la linea deve essere mantenuta al livello logico 1. Ciascuno di questi n intervalli viene chiamato bit di stop. Nell'esempio di *figura 7.7*, la linea viene mantenuta al livello alto per almeno un intervallo n: perciò viene utilizzato un bit di stop. Il codice dati ad 8 bit richiede dunque 10 bit per ogni carattere quando sono inclusi il bit iniziale e il bit di stop. La velocità espressa in baud differisce da quella in bit in molti casi perché nella prima non sono inclusi i bit iniziali e di stop.

Parità

Per il controllo degli errori viene anche comunemente usato il semplice metodo del bit di parità. Parità significa che viene aggiunta al codice un'informazione per aiutare a capire se il codice trasmesso è stato ricevuto correttamente. In questo caso si aggiunge al codice un altro bit. Questa viene chiamata parità a bit singolo. È possi-

Figura 7.8. Codice standard americano per lo scambio d'informazioni (Codice ASCII)

POSIZIONE DEI BIT:											
6				0	0	0	0	1	1	1	1
5				0	0	1	1	0	0	1	1
4				0	1	0	1	0	1	0	1
3	2	1	0								
0	0	0	0	NUL	DLE	SP	0	@	P		p
0	0	0	1	SOH	DC1	|	1	A	Q	a	q
0	0	1	1	STX	DC2	"	2	B	R	b	r
0	0	1	1	ETX	DC3	#	3	C	S	c	s
0	1	0	0	EOT	DC4	$	4	D	T	d	t
0	1	0	1	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	ACK	SYN	&	6	F	V	f	v
0	1	1	1	BEL	ETB	'	7	G	W	g	w
1	0	0	0	BS	CAN	(	8	H	X	h	x
1	0	0	1	HT	EM	)	9	I	Y	i	y
1	0	1	0	LF	SUB	*	:	J	Z	j	z
1	0	1	1	VT	ESC	+	;	K	[	k	{
1	1	0	0	FF	FS	,	<	L	\	l	¦
1	1	0	1	CR	GS	-	=	M	]	m	}
1	1	1	0	SO	RS	.	>	N	^	n	~
1	1	1	1	S1	US	/	?	O	—	o	DEL

Un bit di parità viene aggiunto durante la trasmissione per individuare gli errori. Nella parità pari il suo valore è un 1, quando necessario, per rendere pari il numero di 1 presenti nella parola. Per la parità dispari il totale dovrà essere dispari.

Perché le comunicazioni seriali siano affidabili, il trasmettitore e il ricevitore devono utilizzare lo stesso numero di bit di dati per parola, la stessa condizione di parità, lo stesso numero di bit di stop e la stessa velocità di trasmissione e ricezione; anche il codice per i caratteri e il controllo deve essere identico per entrambi.

bile utilizzare la parità pari o quella dispari purché il trasmettitore e il ricevitore siano concordi. Nel caso di parità pari il bit di parità sommato è uno 0 o un 1 per fare in modo che nel codice sia presente un numero pari di 1. Quando ogni codice di carattere è pronto per la trasmissione, speciali circuiti nel trasmettitore aggiungono il bit di parità così che si abbia sempre un numero pari di uno nel codice del carattere che viene trasmesso. Nel caso di parità dispari il bit di parità aggiunto viene reso 0 o 1 da circuiti speciali in modo che sia presente nel codice del carattere trasmesso un numero di 1 dispari. Se la parità ricevuta non corrisponde, il ricevitore sa che probabilmente c'è un errore nei dati.

Per stabilire comunicazioni affidabili utilizzando la trasmissione seriale dei dati di *figura 7.7*, sia il trasmettitore che il ricevitore devono utilizzare lo stesso numero di bit per dato, la stessa condizione di parità (pari, dispari o nessuna), lo stesso numero di bit di stop e la stessa velocità di trasmissione e ricezione. I dati vengono solitamente trasmessi sotto forma di carattere di macchina per scrivere e di codici di controllo che utilizzano il codice standard americano per lo scambio di informazioni (A-SCII da American Standard Code for Information Interchange). Questo codice binario a 7 bit è riassunto nella tabella di *figura 7.8*; si tenga conto del fatto che il bit 7 è quello più significativo. Se vengono trasmessi 8 bit il bit 8 è sempre 0 (a meno che non sia usato per altro scopo): perciò il codice per la lettera A è 01000001 oppure 41H. Analogamente il codice per il tasto di controllo di ritorno carrello (CR da carriage return) è 00001101 oppure 0DH. Come esercizio il lettore dovrebbe controllare questi codici nella tabella di *figura 7.8*.

Nella trasmissione dei dati vengono usati altri formati e codici; comunque i concetti fondamentali restano quelli appena descritti. Oltre agli standard per codici, devono esserci standard per i livelli di segnale e relazioni temporali precise dei segnali. Uno standard comunemente usato, chiamato protocollo, è lo standard EIA RS232.

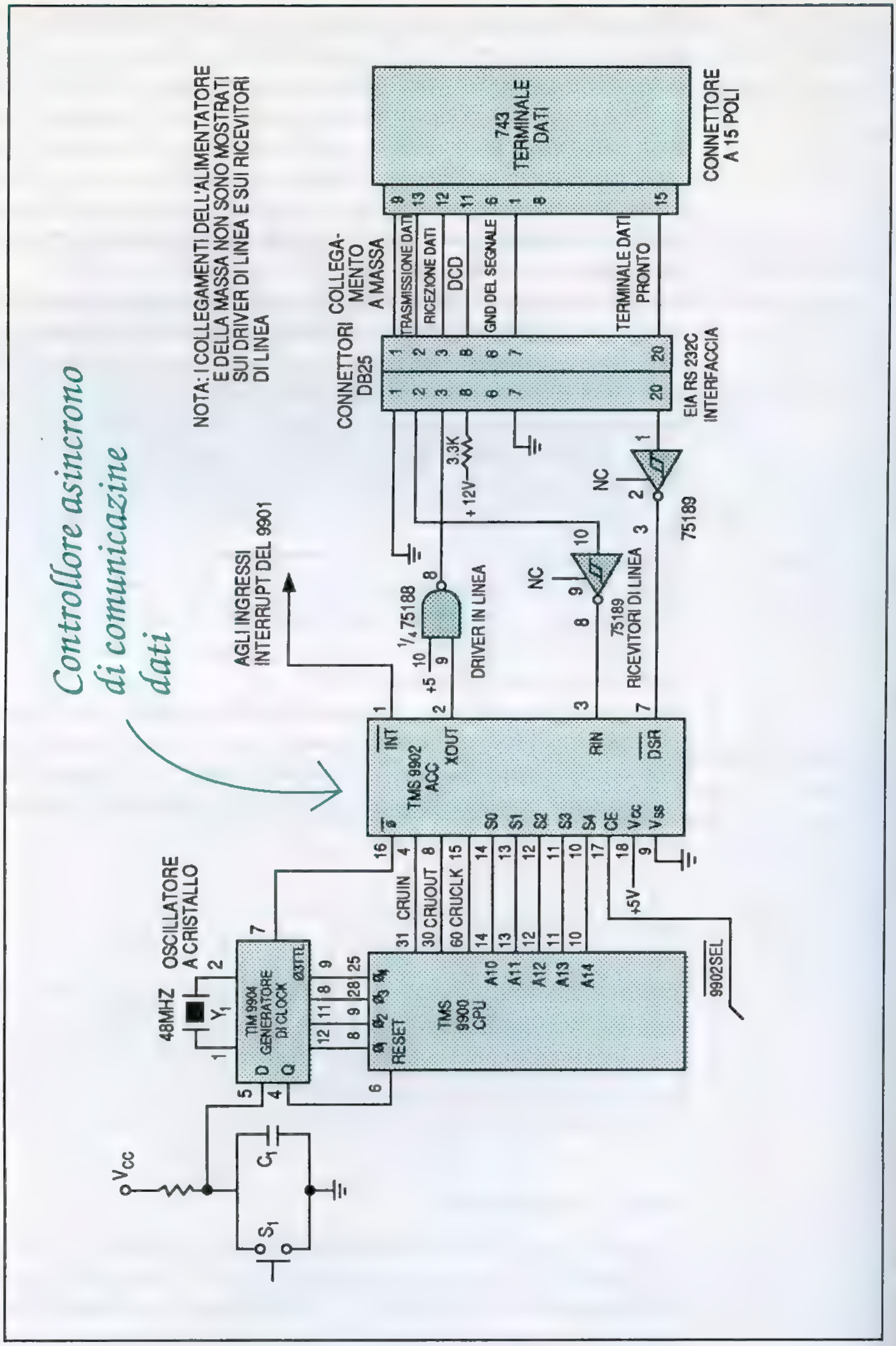
Standard EIA RS232

Electronics Industry Association (EIA) ha introdotto lo standard RS232 per la connessione tra le unità di un computer. La cosa più importante è che lo standard definisce lo 0 logico come una tensione compresa tra -3 volt e -15 volt e l'1 logico come una tensione compresa tra +3 volt e +15 volt. Esso inoltre limita la resistenza che può essere posta su una linea e specifica il piedino di connessione per ogni segnale.

Esempio di interfaccia

I circuiti di interfaccia normalmente disponibili, noti come driver di linea e ricevitori di linea RS232, vengono utilizzati per convertire i segnali digitali del computer nelle condizioni dei segnali RS232. L'interconnessione di base tra il terminale e il computer che utilizza il circuito di interfaccia RS232 è mostrata in *figura 7.9*. Questa figura mostra il circuito di interfaccia da un microprocessore alla linea di dati seriale che usa una speciale unità di controllo chiamata controllore per comunicazioni asincrone (ACC).

Figura 7.9. Interfaccia a un terminale dati 743 con un ACC TMS9902


NOTA: I COLLEGAMENTI DELL'ALIMENTATORE E DELLA MASSA NON SONO MOSTRATI SUI DRIVER DI LINEA E SUI RICEVITORI DI LINEA
743 TERMINALE DATI
CONNETTORE A 15 POLI
CONNETTORI DB25
COLLEGAMENTO A MASSA
TRASMISSIONE DATI
RICEZIONE DATI
DCD
GND DEL SEGNALE
TERMINALE DATI PRONTO
EIA RS 232C INTERFACCIA
3.3K
+12V
NC
75189
AGLI INGRESSI INTERRUPT DEL 9901
1/4 75188
+5
DRIVER IN LINEA
75189 RICEVITORI DI LINEA
INT
TMS 9902 ACC
XOUT
RIN
DSR
S0
S1
S2
S3
S4
CE
Vcc
Vss
+5V
9902SEL
OSCILLATORE A CRISTALLO
48MHZ
Y1
TIM 9904 GENERATORE DI CLOCK
CRUIN
CRUOUT
CRUCLK
TMS 9900 CPU
RESET
A10
A11
A12
A13
A14
VCC
C1
S1

Controllore asincrono di comunicazine dati

Trasmissione

Comunemente i dati vengono trasferiti da un sistema digitale ad un altro nel formato RS232 utilizzando un controllore asincrono di comunicazioni.

L'ACC riceve i dati dal microprocessore un byte alla volta in parallelo o un bit alla volta in serie fino ad avere a disposizione un byte per la trasmissione. Nell'esempio di *figura 7.9*, il dato viene trasferito dal microprocessore all'ACC un bit alla volta finché il processore utilizza un sottosistema I/O a mappa di bit (si ricordino le *figure 7.1* e *7.2*). Una volta che il byte è all'interno dell'ACC pronto per essere inviato, l'ACC genera la sequenza di segnali del tipo mostrato in *figura 7.7* sulla linea XOUT. Questa sequenza viene inviata tramite un driver di linea RS232 al terminale dati attraverso il piedino 3 dell'interfaccia RS232.

Ricezione

Dal ricevitore, dopo che un byte di dati è stato ricevuto ed è stato controllato se il bit di parità è corretto, il dato viene inviato al microprocessore.

Il terminale dati invia un segnale seriale RS232 attraverso il piedino 3 del connettore DB25 al computer. Il segnale viene convertito in livelli di segnale digitale dal ricevitore (75189 in *figura 7.9*) e il segnale viene inviato alla linea RIN dell'ACC. L'ACC riceve questo dato un bit alla volta secondo il formato di *figura 7.7*, finché accumula un byte di dati come indicato dalla ricezione del bit di stop. Esso inoltre controlla se ci sono errori nel bit iniziale, nel o nei bit di stop e nel bit di parità (se c'è). Una volta ricevuto un byte il processore viene avvertito tramite la linea di interruzione. A questo punto il processore può inviare un segnale per chiedere all'ACC di determinare se sono stati trovati degli errori nella trasmissione. Se non ci sono errori esso invia un segnale all'ACC per dirgli di trasmettere i dati al processore.
Oltre alle normali linee seriali di ingresso (RIN) e di uscita (XOUT) ci sono altri segnali di protocollo come RTS (Request To Send, cioè richiesta di trasmettere) e CTS (Clear To Send, cioè azzerare per trasmettere) che non sono mostrati in *figura 7.9*. Un segnale di questo tipo, mostrato nel blocco ACC di *figura 7.9*, è il DSR (Data Set Ready Not), segnale che notifica all'ACC il fatto che il terminale dati è in linea ed è pronto per scambiare informazioni.
Mentre i circuiti rappresentati dai blocchi di *figura 7.9* possono essere molto complessi internamente, i collegamenti fra i blocchi sono relativamente semplici. Questo semplifica la ricerca e l'eliminazione dei guasti come l'esempio seguente illustrerà.

Problemi principali del terminale

Le comunicazioni tra un computer e un terminale possono venir meno a causa di uno di questi tre problemi generali:

1. Il terminale può essere guasto, nel qual caso gli sforzi per determinare il guasto dovranno essere diretti verso la circuiteria all'interno del terminale. Questa ricerca può essere molto complicata.

2. I circuiti del computer quali l'ACC, i driver o i ricevitori di linea o anche il processore possono essere guasti, nel qual caso dovranno essere utilizzati i metodi di ricerca ed elimi-

nazione dei guasti discussi per i circuiti delle *figure* dalla *7.1* alla *7.6*.

3. Il cablaggio tra il terminale e il computer può essere guasto. In questo caso le semplici tecniche di misurazione di resistenza descritte nel capitolo 3 possono essere sufficienti per localizzare e correggere il problema.

Come problema esemplificativo considereremo la configurazione di *figura 7.10*. Questo caso realmente verificatosi servirà per passare in rassegna tutti gli approcci fondamentali per la ricerca dei guasti in un qualsiasi problema, inclusi quelli che richiedono una strumentazione digitale relativamente complessa. Tre uffici erano stati collegati in modo tale che un terminale posto in ciascun ufficio potesse essere collegato ad un unico cavo che andava ad un computer centrale collocato ad una certa distanza dai terminali stessi. Ogniqualvolta un ufficio voleva accedere al computer, la spina del cavo centrale veniva inserita nella presa per il cavo che andava a quell'ufficio. All'altro capo del cavo di ciascun ufficio c'era un connettore RS232 DB25 montato in una scatola di derivazione. Un connettore RS232 e un cavo collegavano questa scatola di derivazione col connettore RS232 sul terminale. Il terminale era costituito da un'unità CRT con un'interfaccia RS232 e da un'unità di tastiera separata e collegata con il monitor tramite un cavo di interfaccia di dati in parallelo.

In molti casi i possibili problemi nel cablaggio dei terminali possono essere isolati sostituendo i terminali stessi.

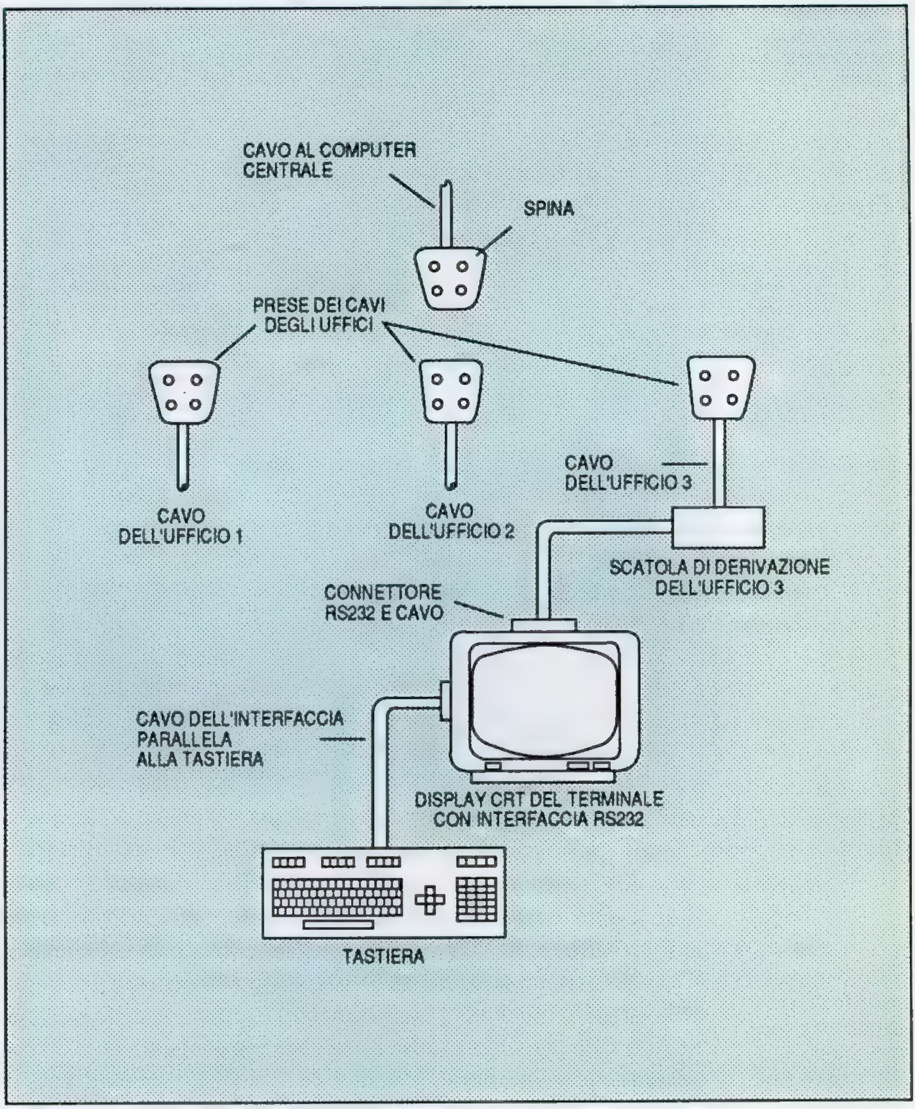
Dopo che furono installati i cavi e i terminali, si scoprì che il terminale nell'ufficio 3 si accendeva ma non comunicava col computer. I terminali negli uffici 1 e 2 funzionavano correttamente e trasmettevano e ricevevano dati dal computer centrale quando le loro linee erano collegate con la linea centrale. Il terminale nell'ufficio 3 fu portato nell'ufficio 2 e collegato: in questo modo il suo funzionamento era soddisfacente. Allora il terminale dell'ufficio 2 venne provato nell'ufficio 3 e si scoprì così che non funzionava. Perciò la connessione dall'ufficio 3 alla presa della linea centrale doveva essere in qualche modo guasta.

Fu usato un ohmetro per controllare la continuità di ciascun collegamento fra la spina centrale e l'estremità del cavo RS232 e per controllare se si erano verificati dei corto circuiti tra i fili. I quattro fili coinvolti nel collegamento (vedi *figura 7.9*) erano quello di collegamento a massa del segnale, quello di trasmissione dati, quello di ricezione dati e quello di terminale dati pronto. Comunque tutti i collegamenti risultarono corretti e non si scoprirono corto circuiti.

A questo punto sembrava che tutti i componenti del sistema fossero stati controllati e che fosse stato verificato il loro soddisfacente funzionamento; comunque il terminale continuava a non comunicare con il computer centrale dall'ufficio 3. Sembrava che non ci fossero altri test che potessero servire per determinare il guasto, però doveva esserci una ragione che impediva al terminale dell'ufficio 3 di funzionare. Voi che cosa avreste fatto?

Era chiaro che il problema doveva trovarsi nel sistema di cavi e che o veniva trovato il collegamento guasto o dovevano essere sostituiti tutti i cavi dell'ufficio 3. Il riparatore ricordò che il controllo della continuità e dell'eventuale corto circuito era stato effettuato con la tensione in corrente continua dell'ohmetro, ma che il segnale dati era un segnale di impulso di 1200 bit al secondo. Poiché la frequenza dell'im-

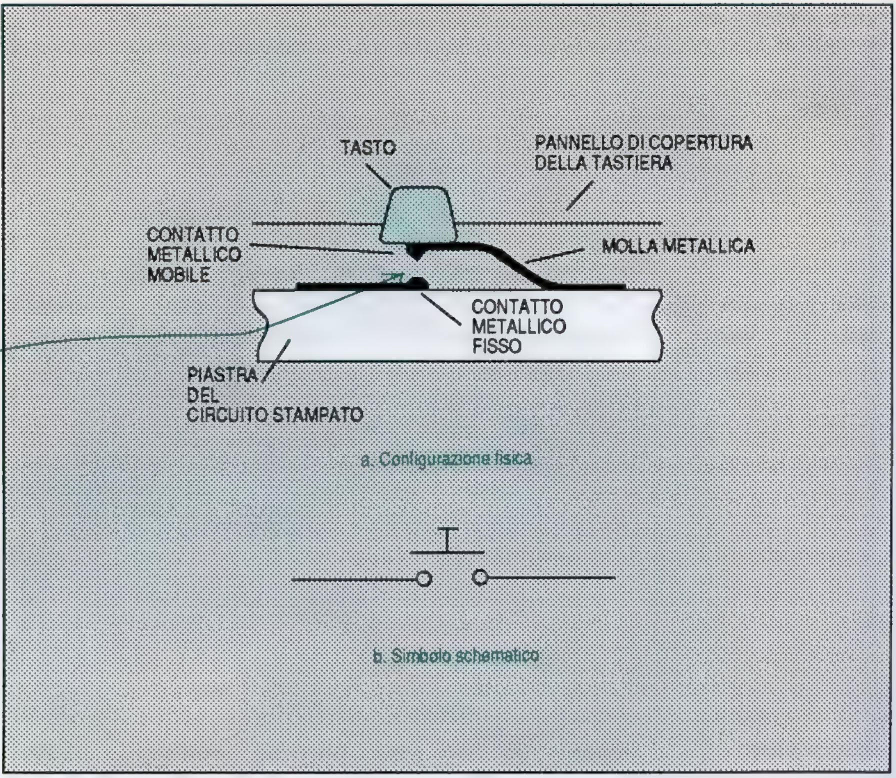
Figura 7.10. Esempio di ricerca dei guasti in un terminale


CAVO AL COMPUTER CENTRALE
SPINA
PRESE DEI CAVI DEGLI UFFICI
CAVO DELL'UFFICIO 1
CAVO DELL'UFFICIO 2
CAVO DELL'UFFICIO 3
SCATOLA DI DERIVAZIONE DELL'UFFICIO 3
CONNETTORE RS232 E CAVO
CAVO DELL'INTERFACCIA PARALLELA ALLA TASTIERA
DISPLAY CRT DEL TERMINALE CON INTERFACCIA RS232
TASTIERA

pulso era elevata e le transizioni da 0 a 1 e da 1 a 0 avvenivano rapidamente, si pensò che un collegamento ad alta resistenza potesse essere la causa del problema. Nel tentativo di localizzare il collegamento guasto furono effettuati ancora i controlli con l'ohmetro sul cablaggio, ma questa volta furono notate le effettive letture di resistenza. Tutti i collegamenti, eccetto uno, avevano una resistenza nella gamma da 1 a 2 ohm; l'unica eccezione aveva 25 ohm. Si trattava della linea dall'uscita del driver di trasmissione alla scatola di derivazione del terminale. Mentre una resistenza di 25 ohm in corrente continua non faceva differenza nel sistema sotto test, la resistenza poteva essere molto più elevata ad alte frequenze. Dopo che furono saldati nuovamente tutti i collegamenti nella linea ad alta resistenza, la resistenza di questa

Figura 7.11. Interruttore di una tastiera tipica

Ingresso indicato dai contatti chiusi


TASTO
PANNELLO DI COPERTURA DELLA TASTIERA
CONTATTO METALLICO MOBILE
MOLLA METALLICA
CONTATTO METALLICO FISSO
PIASTRA DEL CIRCUITO STAMPATO

a. Configurazione fisica

b. Simbolo schematico

linea risultò essere nella gamma da 1 a 2 ohm; perciò l'operazione fatta aveva avuto il suo effetto. Il terminale fu collegato e funzionò correttamente.

Nell'esempio, una connessione saldata con poca spesa non permetteva il corretto funzionamento di una parte molto costosa della strumentazione.

Una connessione saldata male aveva dunque impedito a un componente del costo di 600 dollari di funzionare correttamente. Il problema della saldatura della connessione fu risolto perché essa aveva una resistenza leggermente diversa dagli altri collegamenti nel cavo. Il problema non sarebbe stato probabilmente individuato se il riparatore non avesse pensato a tutte le possibili cause e non avesse ricontrollato il cablaggio.

Se il guasto fosse stato all'interno del terminale, allora si sarebbe dovuto inviare il terminale ad un laboratorio di riparazioni, dove tecnici esperti e con a disposizione la strumentazione necessaria avrebbero potuto ripararlo. Comunque alcuni semplici problemi possono essere trovati anche con una strumentazione limitata se il riparatore conosce il funzionamento dello schermo video e della tastiera del terminale.

Alcuni terminali usano una stampante come unico dispositivo di uscita, mentre altri utilizzano una stampante insieme ad un terminale video. Le stampanti presentano problemi di funzionamento diversi da quelli incontrati nei terminali video. Alcuni problemi che si verificano nelle tastiere, nei terminali video e nelle stampanti verranno considerati nella restante parte di questo capitolo.

FUNZIONAMENTO DELLA TASTIERA E PROBLEMI AD ESSO RELATIVI

Le tastiere sono formate da contatti a pulsante e essi sono i componenti che si guastano più facilmente. La *figura 7.11* mostra la struttura fondamentale di un tasto e il suo simbolo schematico. Nella sua forma più semplice e meno costosa il tasto è formato da un contatto metallico fissato su una piastra di circuito stampato e da un contatto metallico mobile che agisce da molla sul tasto. Il secondo contatto potrebbe essere stampato su un foglio di plastica (tastiera a membrana) oppure potrebbe essere una molla metallica inserita nel tasto. In entrambi i casi, premendo il tasto si chiude il circuito elettrico tra il contatto fisso e il contatto mobile. Il passaggio di corrente nel circuito provoca una variazione in un circuito di interfaccia o viene rilevato da un circuito di scansione.

La tastiera è una matrice di contatti metallici fissi su una piastra di circuito stampato che agisce tramite una molla. Tra i problemi più comuni ricordiamo un cattivo allineamento, l'ossidazione e la rottura di un conduttore.

I problemi più comuni che si possono verificare in una tastiera sono il cattivo allineamento dei materiali di conduzione, l'ossidazione della superficie di questi o una rottura in uno dei pezzi metallici. Una rottura o un cattivo allineamento dovrebbero essere evidenti con un semplice esame visivo. Potrebbe essere possibile riparare la rottura con un filo che ripristini i collegamenti laddove la rottura li ha interrotti oppure correggere il cattivo allineamento meccanico piegando uno dei contatti con delle apposite pinze.

Il problema dell'ossidazione può non risultare subito evidente ad un esame visivo. Dato che il metallo è esposto all'ossigeno e all'umidità o ad altri agenti chimici corrosivi, il metallo sulla superficie si trasforma in ossido metallico, che è un isolante. Una tensione bassa come quella usata nei circuiti semiconduttori può non riuscire a far passare la corrente attraverso l'ossido, perciò nessun collegamento può essere effettuato tramite il pulsante. Se i contatti sono facilmente raggiungibili, l'ossido può essere rimosso con della carta vetrata, con una lima, o con una lama affilata.

ATTENZIONE: Per pulire i contatti elettrici non si deve mai usare abrasivo al carborundum dato che le sue particelle sono conduttive e potrebbero provocare un corto circuito.

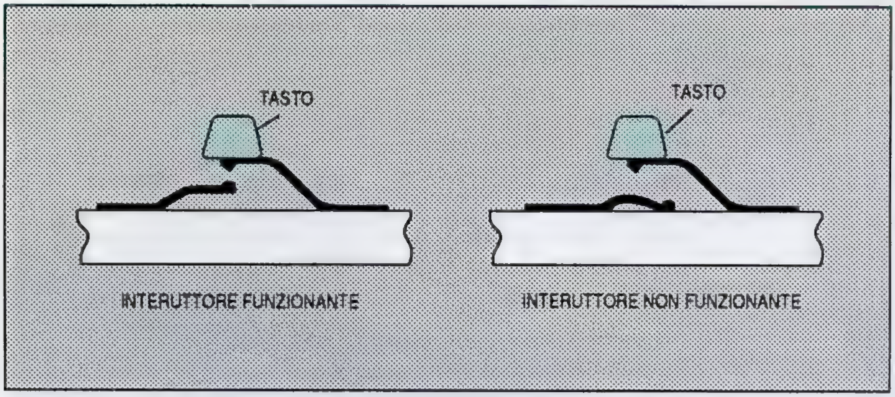
Naturalmente qualunque sia la causa del guasto il metodo di riparazione più sicuro è quello di sostituire il pulsante intero. Sfortunatamente talvolta non è possibile sostituire singoli pulsanti a causa del modo in cui è costruita la tastiera.

Fino ad ora si è supposto che il sistema non rispondesse quando veniva premuto un tasto; comunque in alcuni casi è possibile ottenere una risposta del sistema premendo il tasto molto forte. In tal caso può succedere che il sistema risponda come se il tasto fosse stato premuto più volte rapidamente invece che una volta sola; ciò è dovuto al rimbalzo meccanico del contatto causato dalla forza eccessiva. Se ciò si verifica, la tastiera dovrebbe essere staccata per poter esaminare ed eventualmente pulire i contatti metallici.

Esempio di cattivo allineamento

Un problema di questo tipo, realmente verificatosi, è quello di un calcolatore la cui

Figura 7.12. Esempio di ricerca dei guasti nella tastiera di un calcolatore



tastiera funzionava ad intermittenza. Il calcolatore funzionava correttamente ad eccezione del fatto che non rispondeva correttamente al tasto del numero 8. Premendo questo tasto di solito non si otteneva alcuna risposta, ma premendolo molto forte si provocava la comparsa di parecchi 8 sul display. Questo era molto fastidioso quando si eseguiva la somma di numeri poiché il fatto che l'8 non venisse introdotto o ne venissero introdotti più di uno causava errori e rendeva necessario battere nuovamente l'intera stringa di numeri.

Un allineamento non corretto dei contatti può causare un funzionamento errato.

La tastiera venne aperta rimuovendo alcune viti e i bottoni dei tasti furono staccati dalla matrice metallica a molla del pulsante. Non c'erano né corrosioni né rotture nei componenti di metallo della matrice e non c'erano cattivi allineamenti orizzontali (nel piano della piastra del circuito stampato). Un'esame più attento rivelò però che il contatto inferiore dell'interruttore del numero 8 era più basso di quello degli altri interruttori, come mostrato in *figura 7.12*. Solo se il metallo superiore del tasto malfunzionante veniva spinto il più giù possibile verso la piastra del circuito stampato si poteva ottenere il contatto e ciò era proprio dovuto alla posizione dei contatti. Per il funzionamento corretto degli altri tasti funzionanti era richiesta solo una leggera

Figura 7.13. Esempio di ricerca dei guasti nella tastiera di un computer

Tasto non funzionante	ASCII (Hex)	Carattere sullo schermo	ASCII (Hex)
1	31	9	39
5	35	;	3B
T	54	\	5C
G	47	o	4F

Tasto funzionante	ASCII (Hex)
Z	5A
H	48
Y	59
N	4E

pressione sul contatto superiore. Il rimedio era chiaro: il contatto del bottone doveva essere portato su, alla sua posizione normale. Questo fu fatto utilizzando una fine sonda ad ago per rialzare pian piano il contatto deformato portandolo nella giusta posizione. La tastiera venne reinserita e il tasto 8 riprese a funzionare normalmente, e continuò a farlo finché il calcolatore funzionò.

Esempio di corrosione

L'ossidazione dei contatti richiede che l'operatore alla tastiera debba premere i tasti imprimendo una forza abnorme.

Un altro caso realmente verificatosi mostra un problema che può essere causato dalla corrosione di un contatto. Una tastiera simile a quella di una macchina per scrivere aveva alcuni tasti che dovevano essere premuti parecchie volte, spesso molto forte, perché comparisse il carattere sullo schermo CRT. La copertura del tasto fu rimossa e un'ispezione visiva rivelò che i contatti avevano una leggera coloritura bronzea. Raschiando la superficie del contatto con una lama affilata e passandola poi con un pezzettino di carta vetrata venne ripristinato il luccichio del metallo. Dopo che fu reinserita, la tastiera lavorò normalmente. I contatti dovrebbero essere puliti in questo modo periodicamente (in termini di qualche settimana tra una pulitura e l'altra) per mantenere questo tipo particolare di tastiera perfettamente funzionante. Si trattava di una tastiera relativamente poco costosa che non aveva punti di contatto in grado di resistere alla corrosione. Tastiere più costose prevengono questo problema utilizzando punti di contatto placcati in oro.

Anche i connettori elettrici quali ad esempio quelli di una piastra di circuito stampato o di un cavo che non hanno contatti placcati possono presentare collegamenti intermittenti o connessioni ad alta resistenza. Questi problemi dei connettori possono essere analizzati ed eliminati in maniera molto simile a quella utilizzata per i contatti di una tastiera. Per la maggior parte delle operazioni di pulitura, è disponibile in commercio un solvente che dovrebbe essere utilizzato per i contatti, i connettori per cavo e i connettori a pettine per le piastre di circuito stampato.

Esempio di circuito

I contatti difettosi dei tasti non sono l'unico problema che può provocare il malfunzionamento della tastiera. Potrebbero essere guasti i collegamenti tra la matrice dei pulsanti e il circuito codificatore oppure potrebbe essere difettoso il circuito codificatore stesso. Un esempio di tale problema incontrato realmente dall'autore presentava un sintomo particolare. A volte i tasti nella parte sinistra della tastiera facevano apparire caratteri sbagliati sullo schermo del CRT mentre la parte destra della tastiera continuava a lavorare correttamente. Questo problema era causato da più di un tasto e quindi doveva essere qualche guasto nella parte di circuito comune ai tasti della parte sinistra. Durante un tentativo di individuare il tipo di guasto, furono premuti contemporaneamente alcuni tasti funzionanti insieme ad altri tasti guasti: lo schermo del CRT e i codici ASCII risultanti vennero tabulati come mostrato in *figura 7.13*. L'esame dei codici ASCII per i tasti funzionanti indicava che tutti avevano come cifra esadecimale meno significativa un 8 o un numero maggiore. Per tutti i tasti guasti la cifra esadecimale meno significativa del carattere desiderato era minore di 8, ma il sistema interpretava la chiusura del tasto come un valore ASCII superiore di 8 rispetto al codice corretto. L'unico guasto che poteva aver causato questa situazione era che la cifra binaria D_3 (se D_0 è il bit meno significativo dl codice) fosse bloccata ad 1. Il lettore può verificare questa analisi osservando la tabella ASCII di *figura 7.8*. In questo caso, dunque, è bastato poco lavoro di indagine e di ragionamento per ridurre l'area del guasto ad una linea specifica dell'uscita del circuito decodificatore.

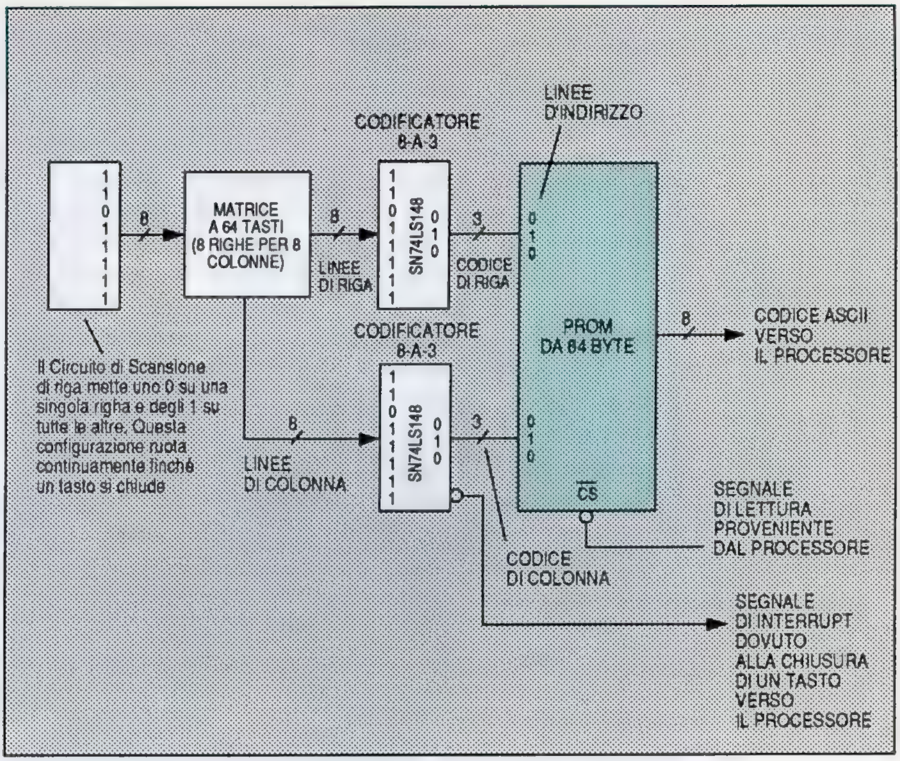
In questa particolare tastiera la codifica era eseguita da una memoria a sola lettura. I suoi ingressi erano il codice di riga e di colonna per il tasto premuto, e la sua uscita era il corrispondente codice ASCII. Il circuito è mostrato in *figura 7.14*. Un circuito di scansione mette uno 0 sulle righe successive di una matrice di interruttori di 8 righe per 8 colonne finché non viene chiuso un interruttore. In quell'istante appare uno 0 sulla linea della riga e sulla linea della colonna che contengono l'interruttore chiuso. Su tutte le altre linee di riga e di colonna c'è invece un 1. Le linee delle 8 righe e delle 8 colonne vanno in due codificatori a priorità 8 a 3 separati le cui uscite a 3 bit sono un codice che rappresenta le linee di riga e di colonna dell'interruttore chiuso. Perciò il codice a 6 bit nella PROM è un indirizzo a 6 bit. Il byte memorizzato nella PROM a quell'indirizzo è il codice ASCII per il tasto premuto.

Poiché si sospettava che l'uscita D_3 della PROM fosse bloccata ad 1, era necessario verificare questo e determinare che cosa stava causando questa condizione. Il corretto funzionamento del codificatore e dei circuiti di scansione fu verificato tenendo premuto un tasto e osservando l'andamento della tensione agli ingressi e alle uscite dei codificatori. Questa operazione venne ripetuta per parecchi tasti differenti, in modo tale da provare il funzionamento di parecchie righe e colonne della matrice degli interruttori.

Come mostrato nell'esempio di *figura 7.14*, venne tenuto premuto un tasto nella seconda riga e nella terza colonna. Le misurazioni confermarono la presenza di uno 0 sulla riga 2 e sulla colonna 3 della matrice. Inoltre, attraverso queste, si ebbe riscontro della correttezza dei codici binari a 3 bit che uscivano dal codificatore per entrare nella PROM. Perciò il circuito di scansione e i codificatori stavano funzionando correttamente. Nonostante questo le misurazioni all'uscita della PROM indicavano che la linea D_3 era bloccata ad 1. Doveva dunque essere guasto il circuito di uscita della PROM sulla linea D_3 oppure il cablaggio verso D_3 doveva aver subito un corto circuito con l'alimentazione a +5 volt.

Il fatto che il problema osservato nel funzionamento fosse intermittente poteva far pensare che il guasto si trovasse nel cablaggio o nei collegamenti di D_3, non nella PROM. Comunque, isolando il piedino di uscita D_3 della PROM dal cablaggio di D_3 si provocò la rimozione dell'1 dalla linea, ma la tensione sul piedino di uscita della PROM per D_3 era ancora bloccato ad 1, il che indicava che la PROM era guasta. La sostituzione della PROM riportò la tastiera al suo normale funzionamento.

Gli esempi trattati in questa parte del capitolo sono relativi a casi reali e rappresentano i problemi che possono verificarsi più comunemente nelle tastiere. Solitamente è ragionevole sospettare che ci siano dei contatti in cattive condizioni o dei guasti nel cablaggio, in quanto è molto più probabile che questi siano causa di un malfunzionamento, piuttosto che si verifichi il guasto di un componente. Comunque, come mostra l'ultimo esempio, anche i componenti possono guastarsi e per individua-


1 1 0 1 1 1 1 1
8
MATRICE
A 64 TASTI
(8 RIGHE PER 8
COLONNE)
8
LINEE
DI RIGA
CODIFICATORE
8-A-3
1 1 0 1 1 1 1 1
SN74LS148
0 1 0
3
CODICE
DI RIGA
LINEE
D'INDIRIZZO
0 1 0
PROM
DA 64 BYTE
0 1 0
CS
8
CODICE ASCII
VERSO
IL PROCESSORE
8
LINEE
DI COLONNA
CODIFICATORE
8-A-3
1 1 0 1 1 1 1 1
SN74LS148
0 1 0
3
CODICE
DI COLONNA
SEGNALE
DI LETTURA
PROVENIENTE
DAL PROCESSORE
SEGNALE
DI INTERRUPT
DOVUTO
ALLA CHIUSURA
DI UN TASTO
VERSO
IL PROCESSORE

Il Circuito di Scansione di riga mette uno 0 su una singola righa e degli 1 su tutte le altre. Questa configurazione ruota continuamente finché un tasto si chiude

Figura 7.14. Esempio di circuito d'interfaccia di una tastiera

re tali guasti è necessario effettuare un controllo completo. Fortunatamente la sostituzione dei componenti guasti è un procedimento semplice che ripristina velocemente il normale funzionamento del sistema.

FUNZIONAMENTO DEL CRT E PROBLEMI RELATIVI

L'unità video di un terminale è un'apparecchiatura piuttosto complessa che normalmente dovrebbe essere riparata da professionisti. Alcuni dei circuiti hanno regolazioni critiche e l'alta tensione sul CRT rappresenta un pericolo sia per le persone che per gli strumenti di test. Comunque alcune ricerche di guasti e riparazioni possono essere compiute anche da un riparatore non professionista. È necessario però fare molta attenzione per evitare di colpire o di graffiare la custodia di vetro.

Fondamentalmente il video è simile ad un televisore in bianco e nero senza i circuiti che selezionano i canali e demodulano la portante. Esso ha dei circuiti che generano i caratteri in risposta al codice ASCII proveniente da una tastiera. I circuiti generatori di caratteri li generano anche quando un codice ASCII viene ricevuto da un'altra fonte, per esempio un computer.

Gran parte dell'analisi funzionale di base che è stata esposta nel capitolo 3 può essere applicata anche qui. I componenti principali del video, compresa un'interfaccia con la tastiera e il connettore seriale RS232, sono mostrati in *figura 7.15*. Gli oscil-

Un controllore video controlla il display del video fornendo gli indirizzi ad una RAM e ad una ROM per i caratteri, i dati di temporizzazione e gli impulsi di blanking e di sincronizzazione per il CRT.

latori orizzontali e verticali e gli amplificatori generano le forme d'onda di deflessione che portano il fascio di elettroni a tracciare una linea orizzontale dopo l'altra sullo schermo del CRT. L'amplificatore video controlla l'intensità del fascio di elettroni nel CRT, ma a differenza della scala di grigi, che variano dal nero al bianco in un televisore, il fascio è acceso o spento. Poiché il fascio è sempre spento ad eccezione di quando appare sul display un particolare carattere, lo schermo del CRT è scuro in assenza di caratteri invece che bianco come nella maggior parte dei televisori. Lo stato del fascio, acceso o spento, dipende dall'insieme di 0 e di 1 inviati dal registro a scorrimento del video. Questo registro viene caricato per ciascuna riga di ciascun carattere dalla ROM che contiene le composizioni dei punti di ogni carattere. L'intero schermo è controllato dall'unità di controllo di temporizzazione del video (VTC da Video Timer Controller) e dai contenuti della RAM del display (video). La RAM del display contiene un codice ASCII per ciascuna posizione sullo schermo. Se è un carattere di spaziatura, quella posizione è scura; in caso contrario, viene mostrato sul display il carattere il cui codice è presente nella RAM.

Il VTC tiene conto di quale riga e di quale posizione di carattere viene mostrata sul display. Esso fornisce questa informazione sulla posizione alla RAM attraverso il codice di posizione e alla ROM del carattere attraverso il codice di riga del carattere. La RAM, a sua volta, fornisce alla ROM del carattere il codice ASCII per il carattere che deve successivamente comparire sullo schermo. Il VTC determina inoltre quando il display sta effettuando un'operazione di ritorno orizzontale o verticale e fornisce gli appropriati impulsi di ritorno e di sincronizzazione agli oscillatori orizzontali e verticali.

Il microprocessore riceve i caratteri dalla tastiera e li invia al computer attraverso l'interfaccia RS232. I caratteri provenienti dal computer e diretti al microprocessore ritornano sulle linee RS232 e aggiornano i contenuti della RAM del display a quella che viene detta posizione del cursore. Il VTC continua a mostrare la vecchia informazione mentre viene acquisita la nuova informazione dalla tastiera o dal computer esterno. Comunque, quando lo schermo è completo, le vecchie informazioni escono dallo schermo (scrolling) a mano a mano che entrano informazioni nuove.

L'alimentazione ad alta tensione fornisce la tensione anodica del CRT. Se si guasta, lo schermo sarà scuro e non comparirà alcun carattere. Se si guasta l'oscillatore verticale, lo schermo mostrerà un'unica linea orizzontale bianca (o una stretta banda di linee). Se si guasta invece l'oscillatore orizzontale lo schermo sarà completamente scuro o potrà presentare una striscia verticale di informazioni sovrapposte. Se l'intero schermo diventa uniformemente illuminato aumentandone la luminosità, ma non contiene caratteri, può esserci un guasto nel VTC, nel processore, nella tastiera, nella RAM, nella ROM o nel registro dei punti. Se i dati non possono essere scritti sullo schermo attraverso la tastiera c'è un guasto nell'interfaccia tra il processore e la RAM o nell'interfaccia tra la tastiera e il processore. Se appaiono sullo schermo caratteri sbagliati quando viene premuto un tasto, il guasto può essere nel circuito della tastiera (come visto precedentemente) oppure nei circuiti della ROM dei caratteri.

Se lo schermo è completamente bianco, la causa più probabile di guasto è da ricercarsi nel collegamento tra la ROM dei caratteri e il registro o nei circuiti tra il registro e l'amplificatore video. È improbabile (ma possibile) che la RAM contenga il co-

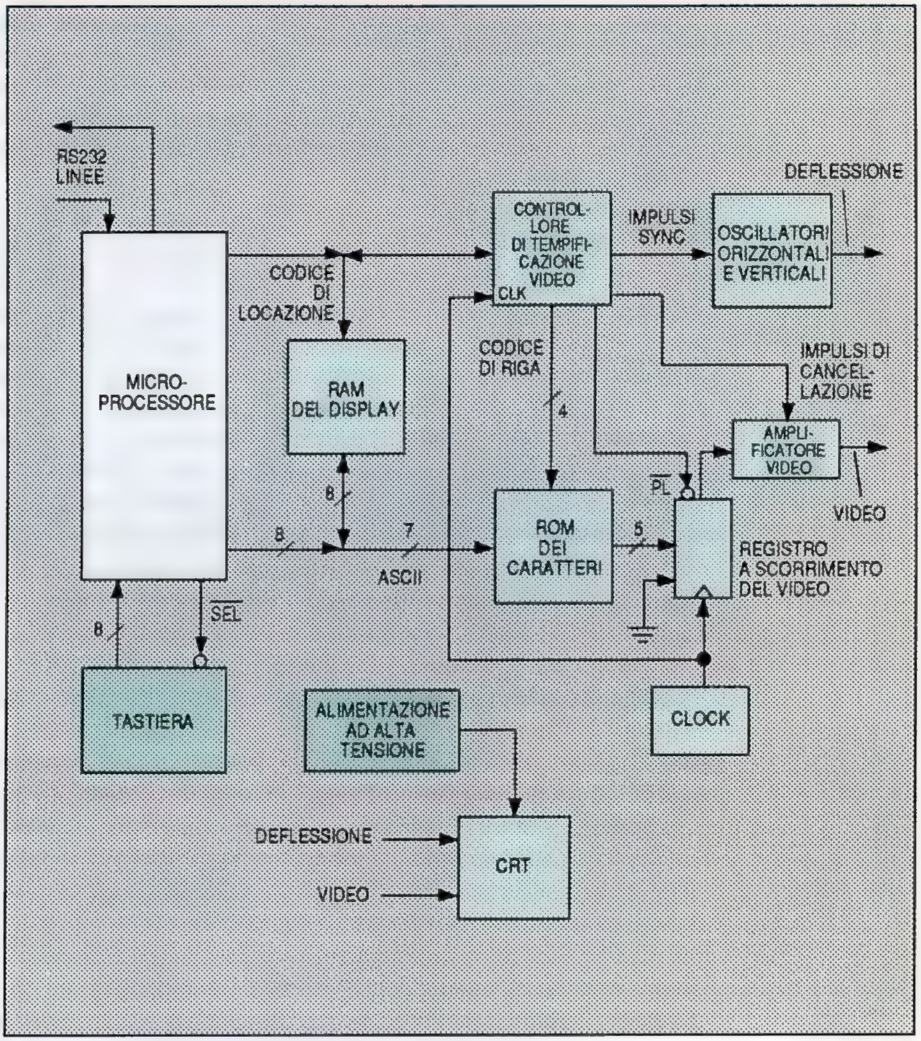

RS232
LINEE
MICRO-
PROCESSORE
CODICE
DI
LOCAZIONE
RAM
DEL DISPLAY
CONTROL-
LORE
DI TEMPIFI-
CAZIONE
VIDEO
CLK
IMPULSI
SYNC
OSCILLATORI
ORIZZONTALI
E VERTICALI
DEFLESSIONE
CODICE
DI RIGA
4
IMPULSI DI
CANCEL-
LAZIONE
AMPLI-
FICATORE
VIDEO
VIDEO
PL
8
8
7
ROM
DEI
CARATTERI
5
REGISTRO
A SCORRIMENTO
DEL VIDEO
ASCII
SEL
8
TASTIERA
ALIMENTAZIONE
AD ALTA
TENSIONE
CLOCK
DEFLESSIONE
VIDEO
CRT

Figura 7.15. Schema a blocchi funzionale di un display CRT

Se non appare nessun carattere sullo schermo, ed esso è bianco, il problema più comune potrebbe essere dato dalla ROM.

dice ASCII per un rettangolo completamente bianco in ogni sua posizione. Inoltre sembrerà che il VTC stia funzionando correttamente. Se la ROM viene disabilitata le sue linee sono fluttuanti (poiché non c'è alcun livello di tensione applicato) o vengono tutte costrette ad assumere il valore 1. Questo fa sì che ci sia il bianco in tutte le aree di display dei caratteri, ma le colonne e le linee tra le posizioni dei caratteri saranno nere. Ne deriva che il display avrà un certo numero di righe e di colonne di rettangoli bianchi. Se l'uscita del registro a scorrimento del video fosse bloccata ad 1 o se l'uscita dell'amplificatore video fosse in corto circuito con l'alimentazione positiva , allora tutti i punti verrebbero accesi e l'intero schermo risulterebbe bianco.

Sarà possibile determinare quale di queste tre condizioni si stia verificando da come apparirà il display (tutto bianco o con rettangoli bianchi in tutte le posizioni dei caratteri) e controllando l'ingresso al circuito dell'amplificatore video. Se il display è costituito da tutti rettangoli bianchi laddove normalmente appaiono caratteri, la ROM è continuamente disabilitata (è possibile che si sia verificata un'apertura nel collegamento di $\overline{CS}$ con la massa) oppure il registro non acquisisce informazioni correttamente (è possibile che si sia verificata un'apertura nel collegamento col terminale di caricamento parallelo $\overline{PL}$).

Se il display è completamente bianco, allora l'uscita del registro a scorrimento dovrebbe essere scollegata dall'ingresso all'amplificatore video. Mettendo a massa l'ingresso dell'amplificatore video lo schermo dovrebbe diventare completamente nero; se ciò non accade c'è un guasto nell'amplificatore video. (Tutto l'amplificatore video dovrebbe essere contenuto in un solo modulo, in modo tale da poter essere sostituito per riparare questo tipo di guasto). Se il display diventa nero quando l'ingresso all'amplificatore video è collegato con la massa, la linea di uscita del registro a scorrimento è bloccata ad 1. Questa uscita dovrebbe essere esaminata per determinare se è messa in corto circuito con l'alimentazione V_{cc} Se questo non avviene il registro stesso dovrebbe essere esaminato per verificare che non sia guasto. Una sequenza simile per la ricerca e l'analisi dei guasti potrebbe essere utilizzata se lo schermo fosse completamente nero, ma si tengano presenti le condizioni menzionate precedentemente che possono causare uno schermo completamente nero.

FUNZIONAMENTO DELLA STAMPANTE E PROBLEMI RELATIVI

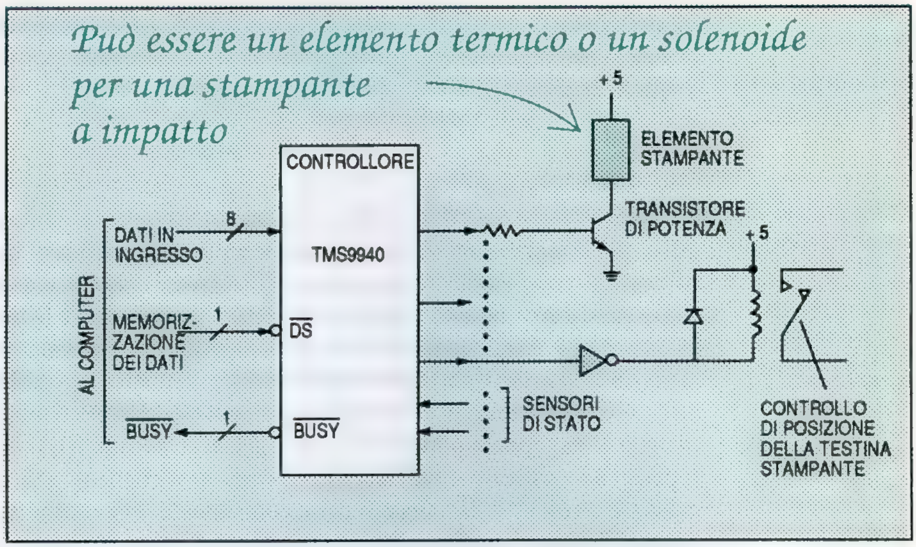
Controllando il livello di tensione di alcune linee, la stampante può comunicare al computer il loro stato, permettendo così al computer di conoscere quando può inviare i dati.

Una stampante può essere un sistema molto semplice o molto complesso, a seconda delle sue caratteristiche e dell'elettronica che la controlla. In *figura 7.16* è mostrato lo schema di una stampante semplice e poco costosa. L'elettronica di controllo consiste di un microcalcolatore a singolo chip (TMS9940) e da circuiti di interfaccia necessari per far muovere gli elementi elettromeccanici e termici del meccanismo della stampante e per ricevere gli ingressi del sensore. La maggior parte delle caratteristiche funzionali del sistema sono immagazzinate nel programma del microcomputer contenuto all'interno del TMS9940. Le comunicazioni tra la stampante e il computer o il terminale avvengono tramite un bus in parallelo ad 8 bit per i codici dati, con un segnale che indica che i dati sono presenti ($\overline{DS}$ in *figura 7.16*) e con un segnale proveniente dalla stampante che indica quando la stampante è pronta per ricevere il carattere successivo ($\overline{BUSY}$ di *figura 7.16*).

I segnali di controllo $\overline{BUSY}$ e $\overline{DS}$ sono segnali attivi bassi e generalmente le loro transizioni sono correlate. Il dato viene posto sulle 8 linee dati dal computer e il segnale $\overline{DS}$ viene mantenuto basso per un periodo di tempo minimo. In un intervallo di tempo specificato, l'unità di controllo della stampante porterà il segnale $\overline{BUSY}$ ad un valore basso per far sapere al computer che la stampante è pronta ad accettare il carattere. Il computer deve conservare il dato sulle 8 linee di ingresso finché il segnale $\overline{BUSY}$ non ritorna di nuovo alto. A questo punto il carattere è stato accettato e memorizzato all'interno di un buffer posto all'interno della stampante ed è pronto per essere stampato. Solo allora il computer potrà mettere un nuovo dato sulle linee ed iniziare un altro trasferimento con un impulso $\overline{DS}$.

L'unità di controllo TMS9940 nell'esempio di *figura 7.16* genera inoltre segnali TTL

Figura 7.16. Schema a blocchi di una semplice stampante



Può essere un elemento termico o un solenoide per una stampante a impatto

per controllare l'elemento stampante attraverso driver di corrente o transistori. L'elemento stampante può essere costituito da solenoidi nel caso di una stampante ad impatto o da contatti termici nel caso di una stampante termica. L'unità di controllo controlla anche il movimento della carta e i meccanismi di posizione della testina scrivente attraverso alcuni relay o attraverso circuiti SCR. Essa riceve anche segnali di stato dai sensori quali OUT-OF-PAPER (Carta finita) e PRINTER NOT READY (Stampante non pronta). La memoria di programma del 9940 contiene lo schema a punti che deve essere utilizzato nella stampa di ogni carattere, così come la memoria della posizione di ogni carattere nella linea. Essa inoltre tiene conto dello stato del buffer, dei tempi necessari per ciascuna operazione (come il ritorno del carrello o l'avanzamento della carta) e assicura che questi tempi vengano mantenuti nei valori corretti.

Ricerca ed eliminazione dei guasti nella stampante

Le stampanti possono avere programmi di controllo diagnostici al loro interno oppure possono essere controllate manualmente. Se si verifica un guasto e i caratteri del test generato internamente alla stampante vengono

Alcune stampanti hanno la caratteristica di possedere un "autotest" che ne controlla il funzionamento; se questo test ha esito positivo, il problema probabilmente è situato nel circuito di interfaccia o nel computer.

Se non si dovesse avere a disposizione l'autotest, il funzionamento della stampante può essere controllato manualmente. Si possono collegare 8 interruttori alle 8 linee dati della stampante. Questi interruttori dovranno essere regolati in modo tale da produrre un qualsiasi codice ASCII. Allora il segnale $\overline{DS}$ può essere generato chiudendo un interruttore che fa generare l'impulso adatto ad un multivibratore monostabile esterno. Se la stampante stampa il carattere impostato con gli 8 interruttori, allora il problema si trova all'interno del computer, del cablaggio o del software.

Una volta noto se il problema è relativo al computer o alla stampante, potranno essere utilizzate le tecniche discusse precedentemente per isolare il guasto. Se il guasto si trova nel computer, allora sarà necessario analizzare i circuiti esemplificativi stampati, il problema si trova probabilmente nel computer.

mostrati nelle *figure* dalla *7.1* alla *7.6*. Se il guasto si trova nella stampante, allora sarà necessario ragionare sul funzionamento della stampante stessa nel modo descritto qui di seguito.

Guasti della stampante

La ricerca dei guasti in una stampante comporta che si determini se il problema sta nella parte meccanica o nella parte elettronica. Se la parte meccanica risulta funzionante il problema è probabilmente situato nel driver e nei circuiti di controllo.

La maggior parte dei guasti della stampante possono essere individuati velocemente. Se, per esempio, la testina stampante resta fissa in una posizione, deve essersi verificata una condizione di blocco a 0 o di blocco ad 1 nel circuito di controllo, oppure si deve essere guastato il relay o il motore che fa avanzare la testina di stampa. Se la carta non avanza quando c'è un ritorno carrello, così che tutti i caratteri risultano stampati uno sopra l'altro su di un'unica linea, si dovrebbe verificare una condizione simile sul circuito di controllo che guida il solenoide o il motore di avanzamento della carta, oppure potrebbero essere guasti il solenoide o il motore.

È possibile verificare il funzionamento di ciascun elemento individualmente utilizzando le tecniche di isolamento di un componente. Per esempio si può isolare l'ingresso al driver e applicargli un 1 per attivare il relay. Se quest'ultimo non funziona, si può isolare l'uscita del driver e mettere a massa direttamente la bobina del relay. Se, anche in questo caso, il relay non si attiva, deve essere guasta la bobina del relay. Se, invece, il relay si attiva, allora sarà probabilmente bloccato ad 1 il circuito di uscita del driver. Se la bobina del relay e il driver funzionano e i contatti del relay si chiudono (cosa che potrà essere determinata con un controllo di continuità), deve esserci un guasto nel circuito del motore o nel solenoide. Ciò può essere provato mettendo un corto circuito tra i contatti del relay per vedere se avviene ciò che dovrebbe. Se non è così, c'è un problema nel cablaggio o nel motore o nel solenoide stesso.

In una stampante a matrice di punti, ciascun carattere è formato da una particolare disposizione di punti. Ciascun punto nella direzione verticale è controllato individualmente. Se il solenoide o il circuito di controllo del solenoide è guasto per un particolare punto, questo punto non verrà stampato in nessun carattere. Il risultato sarà una linea stampata attraversata da una sottile linea bianca. L'unica causa che può provocare questo tipo di guasto è il mancato funzionamento del circuito tra il 9940 e l'elemento stampante di quel punto. Potrebbe essere guasto uno dei transistori del driver, l'elemento stampante o l'uscita dl 9940. È possibile controllare il funzionamento di ciascuno di essi individualmente utilizzando le tecniche di isolamento descritte precedentemente. Perciò ragionando sul funzionamento è in generale possibile individuare con precisione la parte del circuito di *figura 7.16* dove è probabile che si trovi il guasto. È quindi possibile procedere con alcune semplici prove ed esperimenti per determinare con precisione di che problema si tratta.

CHE COSA ABBIAMO IMPARATO?

1. La ricerca dei guasti dei circuiti di ingresso e di uscita non è significativamente diversa da quella degli altri sottosistemi digitali una volta che si sia compreso il funzionamento dei circuiti I/O.
2. Lo stesso vale per i componenti periferici I/O quali i terminali CRT e le stampanti.
3. Generalmente le tecniche fondamentali di esame visivo, controlli di continuità, controlli di tensione, ricerche di corto circuiti e sostituzione di componenti basteranno a determinare la causa del problema.
4. Per alcuni problemi la particolare forma del segnale e la temporizzazione sono critiche: perciò queste caratteristiche del segnale andrebbero controllate. Ciò richiede tecniche di test differenti ed una strumentazione che verrà descritta nei capitoli seguenti.

Quiz del Capitolo 7

1. In un sistema di I/O a mappa di bit:
   a. Ogni bit occupa il proprio indirizzo di locazione di memoria.
   b. Ad ogni byte è assegnato un indirizzo di memoria.
   c. Ad ogni bit è assegnato un indirizzo di I/O.
   d. Ad ogni byte è assegnato un indirizzo di I/O.

2. In un sistema di I/O a mappa di memoria:
   a. Ogni bit occupa il proprio indirizzo di locazione di memoria.
   b. Ad ogni byte è assegnato un indirizzo di memoria.
   c. Ad ogni bit è assegnato un indirizzo di I/O.
   d. Ad ogni byte è assegnato un indirizzo di I/O.

3. In un sistema a mappa di I/O:
   a. Ogni bit occupa il proprio indirizzo di locazione di memoria.
   b. Ad ogni byte è assegnato un indirizzo di memoria.
   c. Ad ogni bit è assegnato un indirizzo di I/O
   d. Ad ogni byte è assegnato un indirizzo di I/O.

4. Nell'esempio di *figura 7.1* il quarto flip-flop corrisponde all'indirizzo di bit:
   a. F0H
   b. 7F3H
   c. 7F4H
   d. 7F8H
   e. 7FBH
   f. 7FCH

5. Nell'esempio di *figura 7.1*, i dati verranno memorizzati dal flip-flop prescelto quando:
   a. I dati sono resi disponibili dal processore.
   b. Le linee di indirizzo sono predisposte secondo un codice appropriato.
   c. Il segnale CRUCLK compie una transizione da un livello basso ad un livello alto.
   d. Il segnale CRUCLK compie una transizione da un livello alto ad un livello basso.

6. Nell'esempio di *figura 7.3* il registro occupa gli indirizzi:
   a. Indirizzo di memoria 1007H.
   b. Indirizzo di registro 1007H.
   c. Indirizzi di bit dal 1007H al 100FH.

7. Nel circuito di *figura 7.5* lo scopo del diodo è:
   a. Di aumentare il tempo di commutazione del relay.
   b. Di diminuire il tempo di commutazione del relay.
   c. Di proteggere la bobina del relay da una tensione di polarità sbagliata.
   d. Di proteggere la porta da una tensione troppo alta.

8. Se è bloccato a 0 il bit D4 di una tastiera codificata ASCII, che cosa viene trasmesso quando si preme il tasto H?
   a. H
   b. X
   c. 8
   d. @

9. Se è bloccato ad 1 il bit D_4 di una tastiera codificata ASCII, che cosa viene trasmesso quando si preme il tasto H?
   a. H
   b. X
   c. 8
   d. @

10. Se i contatti del relay di *figura 7.5* non si chiudono quando il bit di uscita è a 0:
    a. Il circuito del driver di corrente è guasto.
    b. C'è un'apertura nella bobina del relay.
    c. Il diodo è messo in corto circuito.
    d. Tutte le risposte precedenti sono egualmente possibili.
    e. Il circuito funziona normalmente.

PRINCIPALI ASPETTI DELLA TEMPORIZZAZIONE

SCOPO DEL CAPITOLO

La maggior parte dei problemi che sono stati esaminati fino a questo punto era dovuta a corto circuiti o a circuiti aperti nel cablaggio di un sistema digitale o in una delle sue componenti. Tali guasti possono essere rilevati con semplici misurazioni di tensione o di resistenza oppure utilizzando tecniche di rilevazione del segnale. C'è un'altra caratteristica funzionale molto importante che deve essere considerata e che è conosciuta con il nome di temporizzazione. Infatti un circuito non funziona, un segnale non passa attraverso una porta, e non viene nemmeno memorizzato un dato se non si ha un segnale di trigger che dica quando queste operazioni possono essere effettuate - un segnale di temporizzazione, per l'appunto. In questo capitolo verranno considerate tali funzioni di temporizzazione e di controllo. Per determinare i guasti nella temporizzazione è necessario conoscere dettagliatamente i segnali e i loro effetti sul funzionamento del sistema. Dapprima verranno esaminati i segnali di temporizzazione e di controllo dei sistemi sincroni.

LA TEMPORIZZAZIONE NEI SISTEMI DIGITALI

La temporizzazione dell'impulso e la forma dell'impulso sono entrambi importanti nella trasmissione e nell'elaborazione digitale.

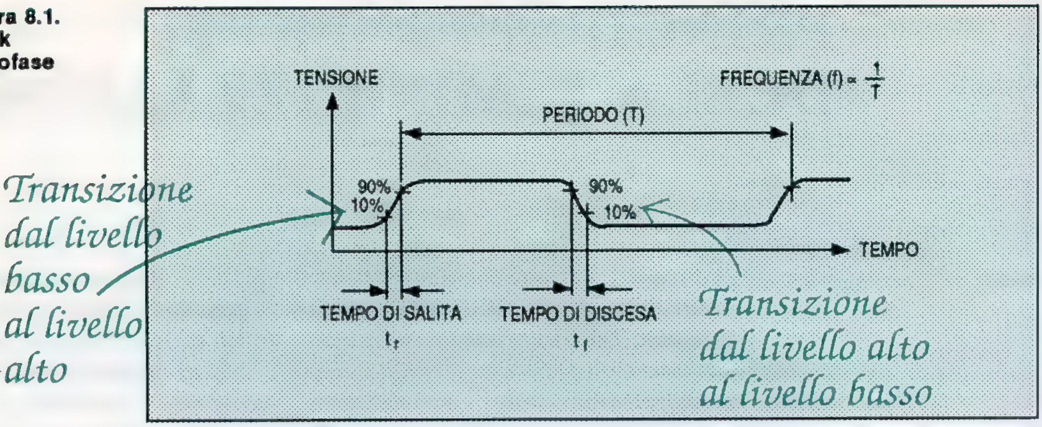
I segnali di temporizzazione in un sistema digitale possiedono determinate caratteristiche se il sistema funziona correttamente. Generalmente queste caratteristiche riguardano il tempo in cui si verificano alcuni segnali in relazione con altri segnali del sistema. Comunque è necessario considerare anche la forma del segnale come una funzione del tempo. Nei sistemi sincroni tutti i segnali di temporizzazione sono derivati da un generatore di sincronismo centralizzato - il clock del sistema.

Clock monofase

La forma d'onda d'uscita del clock dovrebbe avere una transizione dal livello basso al livello alto (tempo di salita) e una transizione dal livello alto al livello basso (tempo di discesa) il più possibile simili ad un'onda quadra.

Alcuni sistemi digitali utilizzano un segnale di clock monofase dal quale derivano tutti gli altri segnali di temporizzazione. Altri sistemi digitali utilizzano invece un clock multifase nel quale due o più segnali di clock fra loro correlati vengono utilizzati per determinare la temporizzazione principale del sistema. Il grafico della tensione in funzione del tempo di un clock monofase è mostrato in *figura 8.1*.
Si ricordi dal capitolo 1 che il tempo impiegato dal segnale per effettuare la transizione dal livello basso al livello alto viene chiamato tempo di salita del segnale. Analogamente il tempo impiegato dal segnale per effettuare la transizione dal livello alto al livello basso viene detto tempo di discesa. Come mostrato in *figura 8.1* questi tempi vengono misurati tra il punto al 10% di ampiezza sull'onda e il punto al 90% sull'onda stessa o viceversa. Questi tempi sono riportati nelle specifiche elettriche di ciascun componente digitale. Nel funzionamento reale i tempi di salita e di disce-

Figura 8.1. Clock monofase



Transizione dal livello basso al livello alto

Transizione dal livello alto al livello basso

sa di un dato componente sono influenzati da fattori quali la lunghezza del cablaggio che accoppia i segnali o dal numero di ingressi che una certa uscita deve pilotare. Un altro tempo caratteristico del segnale di clock è la sua frequenza o periodo. Si ricordi che il periodo T è il tempo che intercorre tra due punti corrispondenti di una certa onda, come mostrato in *figura 8.1*. Esso è uguale a 1/f; perciò la frequenza è il reciproco del periodo (f=1/T). Il periodo dell'onda di clock è determinato dal circuito generatore ed è spesso controllato con estrema precisione da un oscillatore a cristallo. L'oscillatore a cristallo vibra ad alta frequenza e genera gli impulsi che il generatore di clock utilizza per creare il clock del sistema. I segnali di tempificazione devono commutare entro un certo intervallo di tempo altrimenti i circuiti nel sistema non funzioneranno correttamente. Questo intervallo non deve essere troppo lungo, altrimenti componenti come i circuiti di memorizzazione dinamica (si ricordi il ripristino delle RAM dinamiche spiegato nel capitolo 6) perderanno le loro informazioni. D'altra parte l'intervallo non deve essere troppo breve, altrimenti i circuiti nel sistema che hanno una risposta lenta non avrebbero il tempo necessario per rispondere al segnale di clock. I circuiti digitali integrati come i flip-flop, i registri e i contatori hanno una frequenza minima e massima di funzionamento le quali, a loro volta, determinano l'ampiezza degli impulsi di clock accettabili.

La frequenza di clock generata da un oscillatore controllato da un cristallo (cosicché la frequenza è stabile) non deve essere troppo lenta perché altrimenti le DRAM perdono i loro dati.

Clock multifase

Alcuni sistemi digitali, in particolare i microprocessori e i microcomputer, richiedono un segnale di clock a 2 o a 3 fasi. Forme d'onda esemplificative per un clock a 2 fasi sono mostrate in *figura 8.2*. La principale caratteristica di tali forme d'onda è che esse hanno lo stesso periodo (o la stessa frequenza), ma la tensione della fase 1 si trova al livello 1 in tempi diversi rispetto alla fase 2. Ciascuna di esse ha i suoi particolari tempi di salita e di discesa, così come un ritardo fissato tra una forma d'onda e un'altra. Alcuni microprocessori o microcomputer necessitano di un clock a 4 fasi, il quale può avere forme d'onda simili a quelle mostrate in *figura 8.3*.

In alcuni microprocessori e microcomputer i segnali di clock sono generati internamente. In altri casi il segnale è fornito da un generatore di clock esterno che viene ven-

I sistemi digitali dei microcomputer solitamente richiedono un segnale di clock a 2 o 4 fasi. I periodi devono essere identici, ma le fasi sono differenti da un valore fissato e stabile.

Figura 8.2.
Clock a 2 fasi

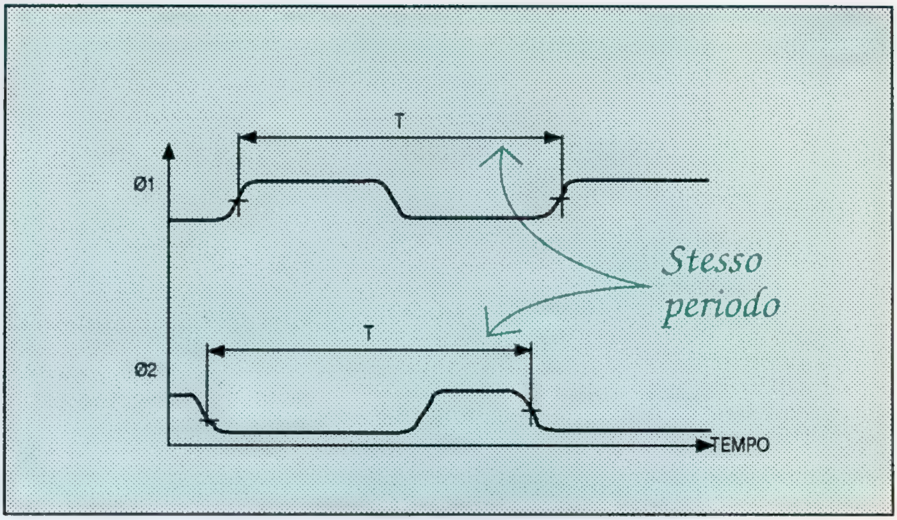
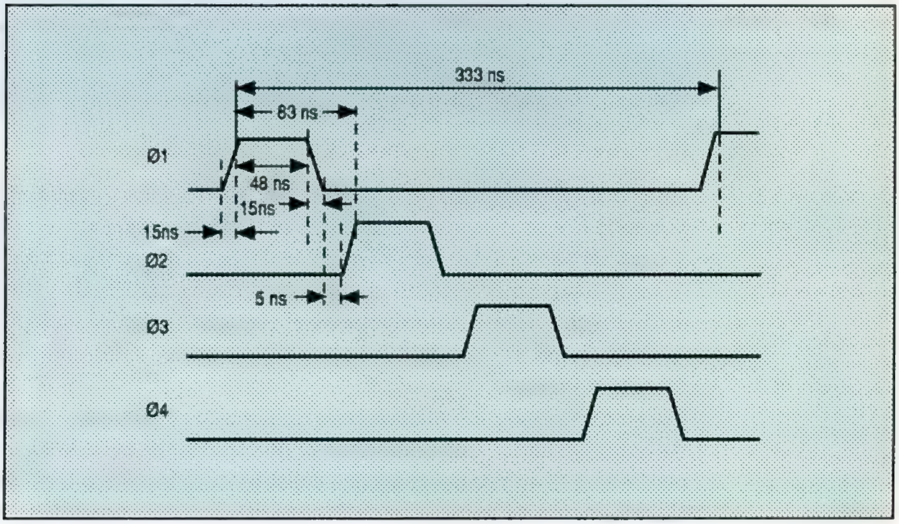


Figura 8.3.
Clock a 4 fasi
(Da: W.D.Simpson, Gerald Luecke, D.L.Cannon and D.H.Clemens, 9900 Family System Design, Texas Instruments Inc., 1978)

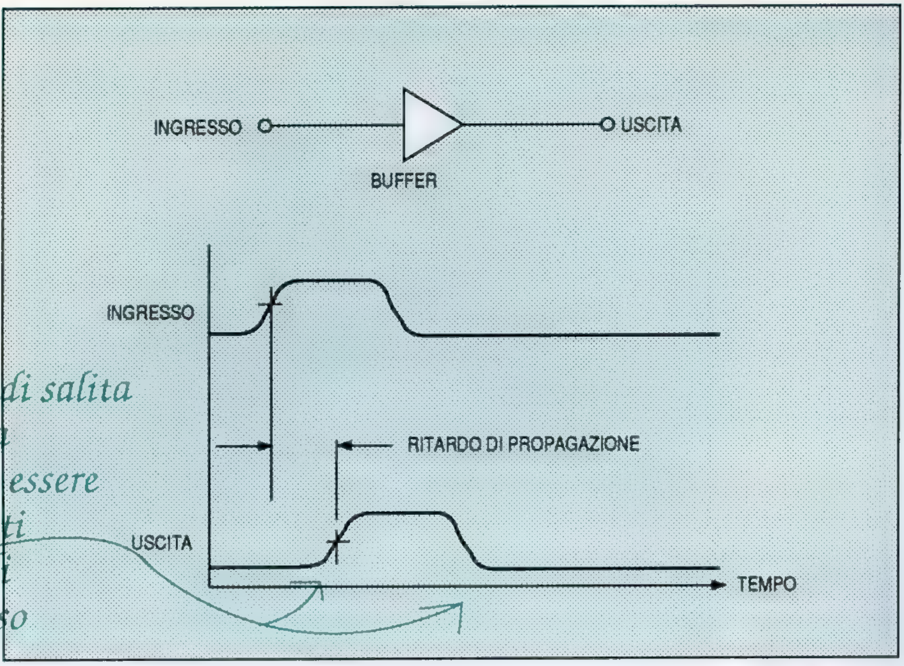


duto dal fabbricante del microprocessore stesso. I generatori di clock multifase, sia interni che esterni ad un microprocessore, vengono solitamente controllati da cristalli, così che il generatore è molto stabile e i segnali di temporizzazione vengono emessi con molta regolarità.

Segnali di temporizzazione

I segnali di temporizzazione in un sistema digitale sincrono possono essere semplicemente una copia della forma d'onda di clock o uno degli impulsi di clock inviati attraverso una porta logica. L'effetto dell'inviare un segnale attraverso una porta di-

Figura 8.4. Ritardo di propagazione



I tempi di salita e discesa possono essere differenti da quelli d'ingresso

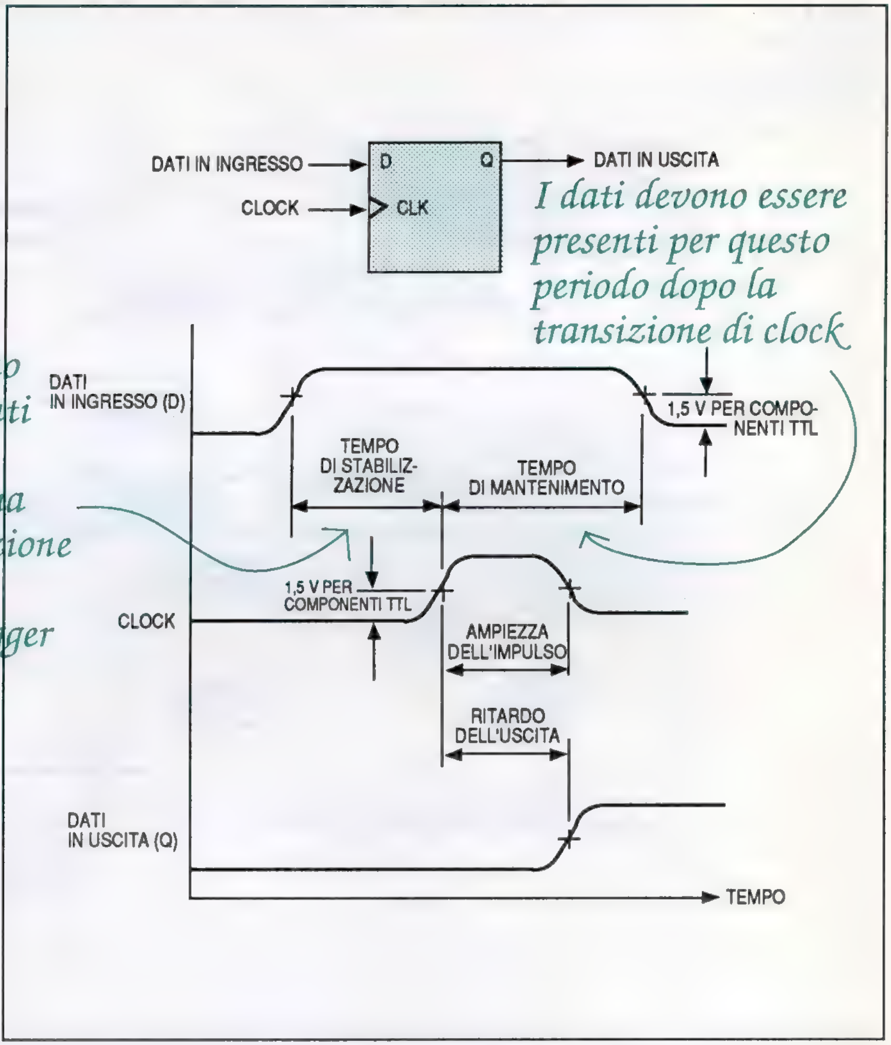
gitale o attraverso un driver è mostrato in *figura 8.4*. La forma d'onda in ingresso, che potrebbe essere il segnale di clock, è mostrata in relazione temporale con la forma d'onda all'uscita del driver. Si noti che il ritardo di propagazione provoca una differenza di tempo tra gli istanti in cui si verificano i due livelli 1 (Si ricordi che il ritardo di propagazione è stato spiegato nel capitolo 1). Anche i tempi di salita e di discesa della forma d'onda in uscita potrebbero essere differenti da quelli della forma d'onda in ingresso.

Quando un impulso logico passa attraverso una porta o un buffer, esso viene ritardato nel tempo (il ritardo di propagazione) e i tempi di salita e di discesa risultano solitamente modificati, soprattutto a causa di un carico aumentato.

Questi tempi vengono misurati con un circuito accoppiato ad un particolare carico. Il circuito di carico può comprendere una resistenza tra la linea di uscita e la tensione positiva di alimentazione e una capacità di carico tra la linea di uscita e la massa. Generalmente all'aumentare della capacità di carico aumentano sia i tempi di salita e di discesa che il ritardo di propagazione del circuito. Il verificarsi di un ritardo di propagazione e di tempi di salita e di caduta eccessivi dovuti ad un componente guasto o a condizioni di carico anomale (una capacità troppo elevata tra la linee di uscita e la massa) può ritardare il segnale oltre il tempo aspettato. Perciò il segnale potrebbe arrivare a destinazione troppo tardi per provocare la corretta risposta del sistema.

Un segnale in ritardo può causare la memorizzazione di dati errati in un flip-flop sincrono o in un registro. I principali requisiti di temporizzazione di tali dispositivi sono stati mostrati in *figura 1.10* e vengono ora ripetuti in *figura 8.5*. I tempi principali sono: il tempo di stabilizzazione (set-up), il tempo di mantenimento e l'ampiezza di impulso del clock. Nell'esempio di *figura 8.5* il fronte di trigger o di memorizzazione del clock è la transizione dal livello basso al livello alto. Il tempo di set-up è

Figura 8.5. Requisiti di temporizzazione per un dispositivo di memoria sincrono


DATI IN INGRESSO
D
Q
DATI IN USCITA
CLOCK
CLK
DATI IN INGRESSO (D)
1,5 V PER COMPONENTI TTL
TEMPO DI STABILIZZAZIONE
TEMPO DI MANTENIMENTO
1,5 V PER COMPONENTI TTL
CLOCK
AMPIEZZA DELL'IMPULSO
RITARDO DELL'USCITA
DATI IN USCITA (Q)
TEMPO

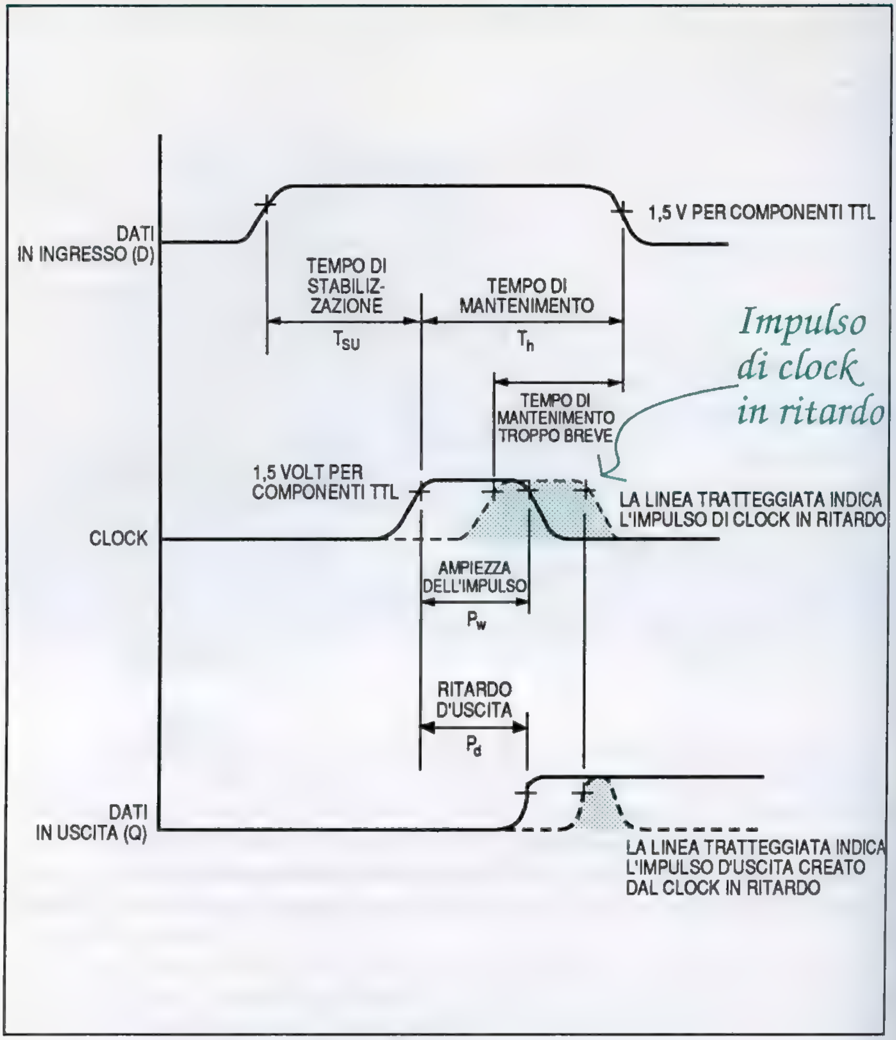
I dati devono essere presenti per questo periodo dopo la transizione di clock

I dati devono essere presenti per questo periodo prima della transizione di clock che dà il trigger

I ritardi di propagazione possono causare un turbamento nell'insieme delle relazioni di temporizzazione tra i segnali nei vari punti del circuito: ne risulta una perdita dell'integrità dei dati.

il tempo entro cui i dati in ingresso devono essersi stabilizzati al loro livello 0 od 1 prima del fronte di trigger del clock. Il tempo di mantenimento è il tempo in cui i dati devono rimanere stabili dopo il fronte di trigger del clock per assicurare la corretta memorizzazione dei dati in ingresso. Molti circuiti hanno un'ampiezza d'impulso di clock che è il minimo necessario per un funzionamento corretto. Come mostrato in *figura 8.5*, questo è il tempo che intercorre fra il punto posto ad 1,5 volt su un fronte dell'impulso di clock ed il punto posto ad 1,5 volt sull'altro fronte dell'impulso. (Alcuni produttori possono specificare il tempo che intercorre tra il punto col 90% di tensione sul fronte di salita e il punto col 90% di tensione sul fronte di discesa per determinare l'ampiezza dell'impulso, sebbene la maggior parte dei manuali sembra che usi il punto al 50% di tensione.)

Figura 8.6. Effetto di un impulso di clock in ritardo sul segnale dei dati in uscita

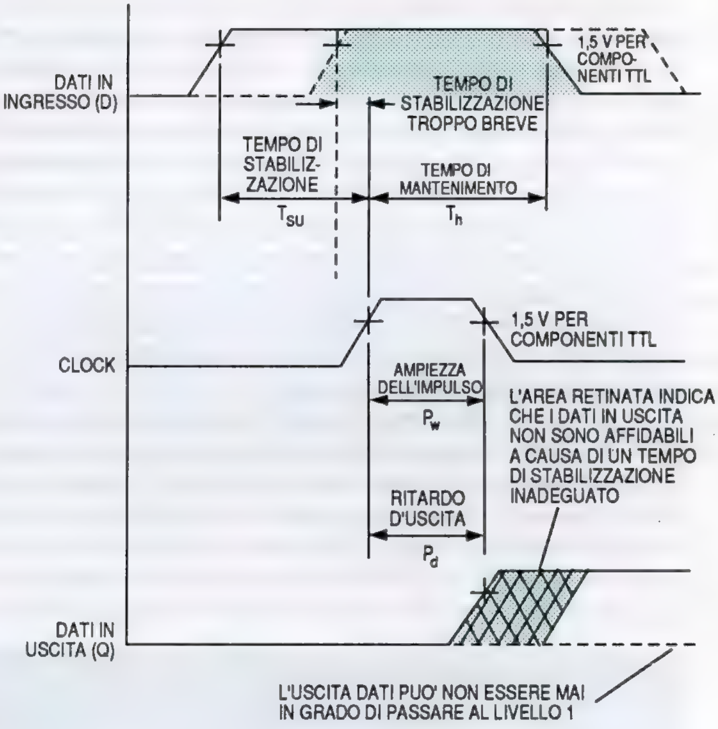

DATI IN INGRESSO (D)
1,5 V PER COMPONENTI TTL
TEMPO DI STABILIZ-ZAZIONE
T_{SU}
TEMPO DI MANTENIMENTO
T_h
Impulso di clock in ritardo
TEMPO DI MANTENIMENTO TROPPO BREVE
1,5 VOLT PER COMPONENTI TTL
CLOCK
LA LINEA TRATTEGGIATA INDICA L'IMPULSO DI CLOCK IN RITARDO
AMPIEZZA DELL'IMPULSO
P_w
RITARDO D'USCITA
P_d
DATI IN USCITA (Q)
LA LINEA TRATTEGGIATA INDICA L'IMPULSO D'USCITA CREATO DAL CLOCK IN RITARDO

Il ritardo di clock può rivelarsi come un tempo di mantenimento o un tempo di set-up inadeguati per i dati che si stanno memorizzando in un flip-flop.

Generalmente una volta che il generatore di clock è stato regolato per fornire l'ampiezza dell'impulso e i tempi di salita e di discesa richiesti da una certa famiglia di componenti logici, i requisiti dell'ampiezza di impulso non costituiscono un problema. Comunque è possibile che il ritardo di un impulso di clock che passa attraverso porte o driver faccia sì che l'impulso di clock arrivi così tardi che i requisiti sul tempo di mantenimento necessari per la corretta memorizzazione dei dati non siano più rispettati. In questo caso non è più assicurato il corretto funzionamento del dispositivo. Il risultato di questo impulso di clock in ritardo è illustrato dalla linea tratteggiata in *figura 8.6*. La linea dati in uscita tenta di mantenersi al livello 1, ma non ci riesce.

Analogamente se i segnali dei dati sono in ritardo rispetto all'impulso di clock, i re-

Figura 8.7. Effetto di un segnale d'ingresso dati in ritardo


DATI IN INGRESSO (D)
1,5 V PER COMPONENTI TTL
TEMPO DI STABILIZZAZIONE TROPPO BREVE
TEMPO DI STABILIZZAZIONE
T_{SU}
TEMPO DI MANTENIMENTO
T_h
CLOCK
1,5 V PER COMPONENTI TTL
AMPIEZZA DELL'IMPULSO
P_w
L'AREA RETINATA INDICA CHE I DATI IN USCITA NON SONO AFFIDABILI A CAUSA DI UN TEMPO DI STABILIZZAZIONE INADEGUATO
RITARDO D'USCITA
P_d
DATI IN USCITA (Q)
L'USCITA DATI PUO' NON ESSERE MAI IN GRADO DI PASSARE AL LIVELLO 1

quisiti sul tempo di set-up non sono più soddisfatti, causando, anche in questo caso, una memorizzazione non affidabile dei dati. Quest'effetto è illustrato dalla linea tratteggiata del segnale in *figura 8.7*. L'area retinata nel segnale di DATA OUT indica che il fronte di salita di questo segnale può verificarsi ovunque in quell'area. Questi problemi sono spesso difficili da individuare e il loro effetto è difficile da osservare. Perciò sono richieste speciali tecniche di misurazione.

L'oscilloscopio è ampiamente usato per visualizzare su di un display e misurare i segnali digitali di un sistema.

LE TECNICHE DI MISURAZIONE

Per determinare i problemi relativi alla temporizzazione si deve innanzi tutto essere in grado di definire la forma dell'impulso di clock e il suo rapporto con gli altri segnali nel sistema. Questi impulsi di clock sono molto veloci e durano solo per un tempo brevissimo. Le frequenze tipiche del clock per microcomputer vanno da 1 a 10 megahertz con periodi di tempo che vanno da 1000 a 100 nanosecondi. I tempi di salita, di discesa, di set-up, di mantenimento e il ritardo di propagazione vanno tutti da 10 a 20 nanosecondi. Perciò è necessario avere a disposizione uno strumento che riesca a visualizzare su un display le variazioni di tensione del segnale in funzione del tempo in questi brevi intervalli. Come menzionato nel capitolo 1, uno strumento capace di effettuare tali misurazioni è l'oscilloscopio. Il funzionamento fondamentale dell'oscilloscopio è trattato nell'appendice A.

Un oscilloscopio a più canali è necessario per confrontare le relazioni temporali tra due o più segnali.

Un oscilloscopio a singolo canale può visualizzare sul display solo un segnale alla volta. Questo è utile per misurare la forma, le caratteristiche temporali e i livelli di tensione di un segnale di clock o di un singolo impulso, ma non sufficiente per misurare le relazioni temporali tra due o più segnali. Con un tale strumento ad un solo canale non sarebbe possibile esaminare i tempi di set-up, di mantenimento o il ritardo di propagazione di due segnali. Per effettuare questo tipo di misurazioni l'oscilloscopio deve essere in grado di visualizzare sul display due segnali contemporaneamente. Questo tipo di oscilloscopio, chiamato oscilloscopio a doppia traccia, fornisce due ingressi di segnale separati ed anche due controlli di calibrazione verticale separati. Esistono inoltre modelli di oscilloscopio a quattro canali che sono necessari per visualizzare sul display segnali del tipo mostrato in *figura 8.3*. Utilizzando uno strumento a più canali le relazioni temporali critiche tra i segnali possono essere esaminate per vedere se avvengono al momento giusto in modo tale da provocare il corretto funzionamento del sistema. Un tipico oscilloscopio a più canali è mostrato in *figura 8.8*. Le forme d'onda che compaiono sul suo schermo sono dello stesso tipo di quelle che il riparatore può incontrare quando utilizza un oscilloscopio per

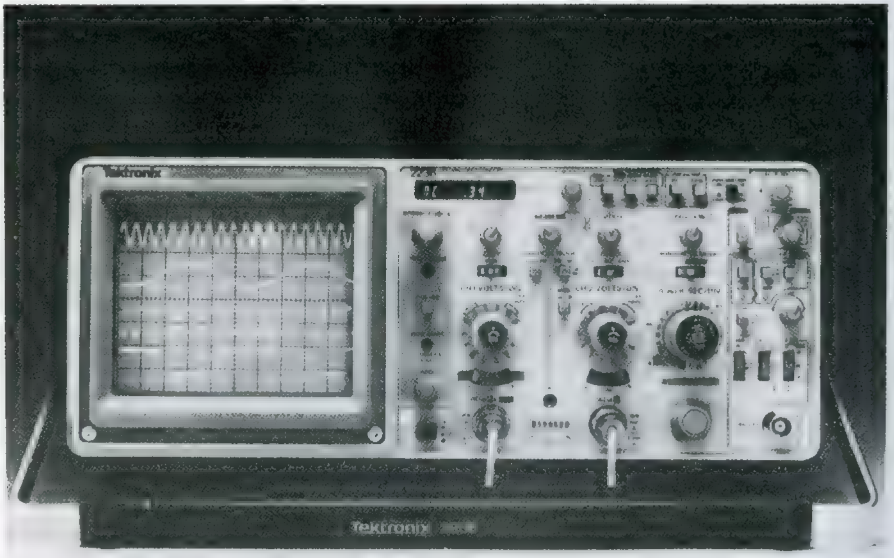


Figura 8.8. Oscilloscopio multitraccia *(Per gentile concessione della Tektronix Inc.)*

isolare un guasto. Con questo tipo di strumento le relazioni temporali che possono essere causa di guasti nel funzionamento di un sistema possono essere verificate.

PROBLEMI DI TEMPORIZZAZIONE NEI SISTEMI SEQUENZIALI

Le relazioni di temporizzazione errate che si verificano in un sistema sono spesso difficili da trovare senza ricorrere a misurazioni particolareggiate dei segnali utilizzando l'oscilloscopio. Generalmente i problemi di temporizzazione in un sistema sincrono sono meno gravi di quelli che si verificano in un sistema non sincrono (asincrono), a meno che il sistema sincrono non operi molto vicino alla frequenza massima tollerata dai suoi componenti. Comunque è possibile che si verifichino errori di temporizzazione in sistemi a bassa frequenza se vengono utilizzati componenti progettati per essere usati ad alta frequenza.

I problemi di temporizzazione in sistemi asincroni possono essere più difficili da isolare di quelli nei sistemi sincroni, a meno che i sistemi sincroni funzionino al limite della loro frequenza.

Gli effetti dei problemi di temporizzazione in un sistema sequenziale possono essere illustrati attraverso l'esempio del distributore automatico citato nel capitolo 5. Lo schema fondamentale del circuito per l'unità di controllo è mostrato in *figura 5.6* mentre i circuiti di ingresso e di uscita associati sono mostrati in *figura 5.3* e nelle *figure* dalla *5.5* alla *5.7*. Il punto di partenza importante per l'analisi dei possibili effetti provocati dai problemi di temporizzazione è il programma di controllo della sequenza o diagramma di flusso. Il programma per questo esempio è dato in *figura 5.8*. Fondamentalmente in un sistema con clock (sincrono) i problemi di temporizzazione nei flip-flop di stato o nel segnale di clock costituiscono le cause più probabili dei guasti nel sistema. Il sistema è protetto dalle transizioni che possono verificarsi tra i fronti di clock (fronti di salita nel caso di *figura 5.6*), in quanto i flip-flop di stato, e di conseguenza le loro uscite, cambiano solo con l'arrivo di un fronte di salita del segnale di clock. Solo se l'ingresso del flip-flop cambiasse nello stesso istante in cui si verifica il fronte del segnale di clock ci potrebbe essere qualche problema. Picchi di rumore che potrebbero indicare erroneamente l'inserimento di una moneta, la selezione di un articolo o un'uscita momentaneamente inaspettata, potrebbero anch'essi costituire un problema: questi però non vengono normalmente considerati come problemi di temporizzazione. Inoltre se la frequenza del clock è sufficientemente bassa, i segnali in uscita saranno presenti sufficientemente a lungo da causare l'operazione desiderata. Nel distributore i segnali VEND, DUMP e CHANGE devono durare abbastanza a lungo da soddisfare i requisiti dei componenti elettromeccanici relativamente lenti che essi controllano.

Guasti nel generatore di clock

Un semplice problema che potrebbe causare un funzionamento inaccettabile del sistema è un guasto al generatore di clock. Se, per esempio, nel circuito di clock mostrato in *figura 5.10a* si verifica un'apertura nel condensatore che determina la temporizzazione per il generatore, il circuito oscillerà alla sua frequenza massima invece che a 100 hertz come desiderato. Questo guasto potrebbe venire facilmente individuato su di un oscilloscopio perché il CRT visualizzerebbe sul display una frequenza di clock eccessivamente elevata. Il funzionamento osservato del sistema, che può far sospettare un guasto nel generatore di clock, potrebbe essere simile al funzionamento provocato da altri guasti che sono stati discussi nel capitolo 5. Perciò è difficile ricavare da un numero limitato d'informazioni la certezza che sia il clock a causare un particolare problema. È necessario quindi compiere ulteriori indagini.

Un effetto che il segnale di clock ad una frequenza molto elevata potrebbe avere nel sistema di *figura 5.6* è che i flip-flop di stato ed altri componenti del circuito potrebbero non essere in grado di rispondere a tale segnale. Se l'ampiezza di impulso del segnale di clock guasto fosse troppo breve per il corretto funzionamento dei flip-flop, questi ultimi non risponderebbero mai ai loro ingressi e la macchina non uscirebbe mai dallo stato 0.

Se si verifica un guasto e il clock funziona ad una frequenza elevata, si ottiene la stessa situazione che si avrebbe se la macchina fosse bloccata nello stato 0 o nello stato 1.

Anche se i flip-flop fossero in grado di compiere la transizione da uno stato ad un altro, le transizioni si verificherebbero così velocemente che l'intera sequenza non durerebbe più di 100 nanosecondi. I dispositivi di uscita del distributore automatico non potrebbero rispondere in un intervallo di tempo così breve e quindi non si avrebbe la restituzione del resto o delle monete e non verrebbero emessi gli articoli.

In ogni caso il distributore automatico si comporterebbe come se fosse spento; comunque questo potrebbe anche essere il caso in cui la macchina resta bloccata nello stato 0 od 1. Perciò devono essere controllati sia il cablaggio di collegamento che il clock.

Una volta accertato che il circuito di clock è la causa del funzionamento (o del mancato funzionamento) della macchina, il problema può essere individuato facilmente. Se, per esempio, il clock emette un segnale ad onda quadra, ma ad una frequenza molto elevata, è possibile verificare se il condensatore è guasto. Questo può essere fatto inserendo temporaneamente un condensatore della capacità di 5 microfarad in parallelo con quello già esistente nel circuito. L'oscilloscopio dovrebbe mostrare una frequenza di clock di circa 100 hertz se il condensatore di clock originale è guasto.

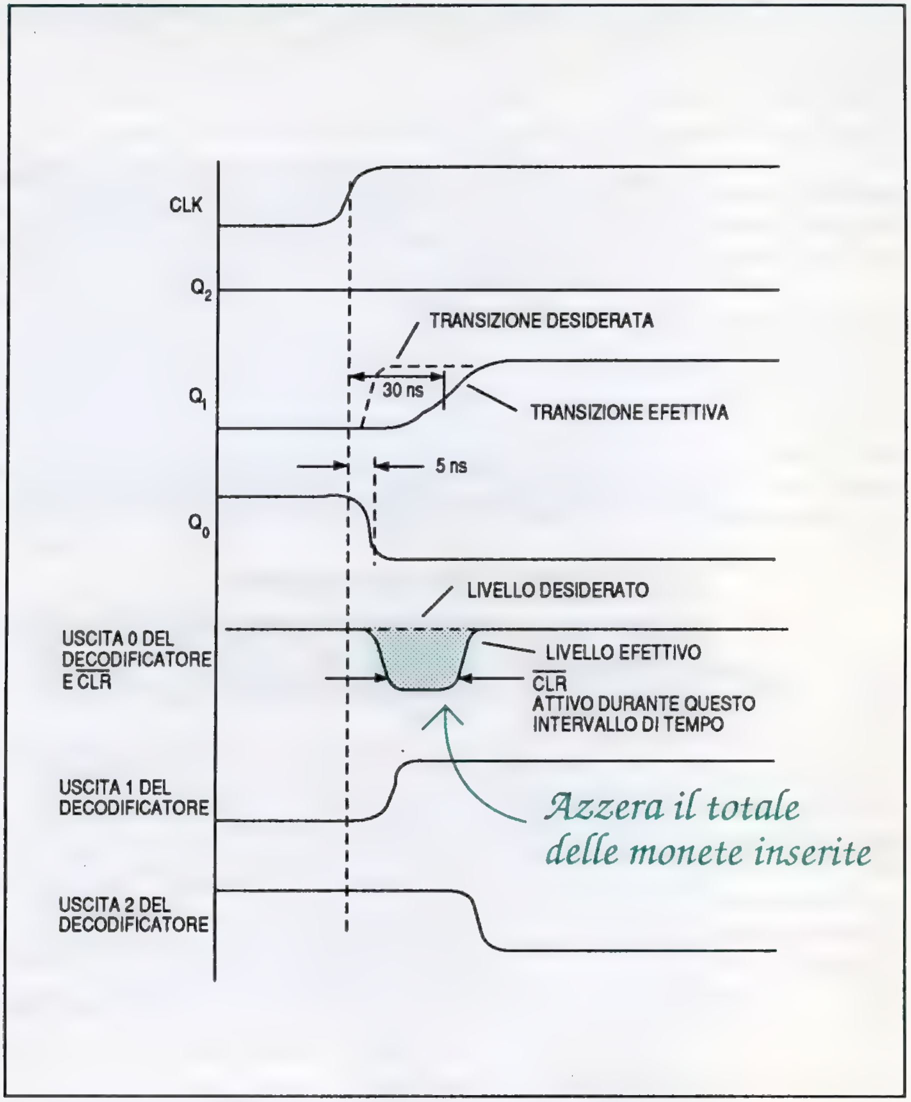
Errata transizione dallo stato 1 allo stato 2

Particolari problemi di temporizzazione possono verificarsi quando due flip-flop stanno cambiando contemporaneamente.

Anche se il circuito di clock funziona correttamente nell'esempio del distributore automatico, è possibile che si verifichi un funzionamento errato dovuto alla temporizzazione del circuito. È possibile, per esempio, che si verifichi un problema quando il codice dello stato successivo differisce da quello dello stato presente in più di una posizione di bit, come nel caso della transizione dallo stato 1 (001) allo stato 2 (010). In questo caso, come mostrato in *figura 5.6*, sia l'uscita Q_0 che l'uscita Q_1 dei flip-flop FF3 e FF2, rispettivamente, devono cambiare in corrispondenza del fronte di salita del segnale di clock. Se Q_0 cambia più velocemente di Q_1 si dovrebbe avere per un istante lo stato 000 tra la condizione dello stato 1 e quella dello stato 2. Ciò viene mostrato in *figura 8.9* insieme con i suoi effetti sui segnali del sistema.

Per qualche motivo Q_0 del flip-flop FF3 inizia la transizione da 1 a 0 prima che Q_1 di FF2 cominci la transizione da 0 ad 1. Ciò può essere dovuto ad un guasto in FF2 che lo rallenta o potrebbe essere dovuto ad un guasto nell'ingresso oppure nel circuito di uscita di FF2. In ogni caso le uscite del flip-flop sono a 000 per un breve periodo, circa 20 nanosecondi come mostrato in *figura 8.9*. Questo provocherebbe a sua volta un'uscita sulla linea $\overline{\text{CLR}}$ che, per questa volta, sarebbe abbastanza lunga da

Figura 8.9. Problema di temporizzazione di un sistema sincrono


CLK
Q_2
TRANSIZIONE DESIDERATA
Q_1
30 ns
TRANSIZIONE EFETTIVA
5 ns
Q_0
LIVELLO DESIDERATO
USCITA 0 DEL DECODIFICATORE E $\overline{CLR}$
LIVELLO EFETTIVO
$\overline{CLR}$ ATTIVO DURANTE QUESTO INTERVALLO DI TEMPO
Azzera il totale delle monete inserite
USCITA 1 DEL DECODIFICATORE
USCITA 2 DEL DECODIFICATORE

cancellare il registro del totale delle monete inserite (*figura 5.5*). Perciò quando la macchina perviene allo stato 010, il valore del totale è 0. È come se non fosse mai stato presente alcun segnale di moneta inserita. Un problema di questo tipo non può verificarsi nelle transizioni dallo stato 2 allo stato 3 o dallo stato 3 allo stato 1 in quanto, in queste transizioni, solo un flip-flop cambia di stato e quindi non importa quanto tempo impiega.

Comunque quando entrambi i flip-flop cambiano, la differenza nella temporizzazione non deve essere troppo grande, pena il verificarsi di un malfunzionamento. In questo caso, risulta azzerato il registro del totale delle monete inserite. Il danno è stato fatto. In *figura 5.6* quando il segnale S è vero, il contenuto del registro del totale è 0 così che il totale è minore del valore dell'articolo e tutte le monete vengono restituite.

Interruzione di una linea

Il comportamento della macchina appena descritto è lo stesso che si osserva quando c'è un'interruzione sulla linea indicata con C così che non si arriva mai agli stati 2 e 3. Comunque il riparatore può determinare quale di questi due guasti sta causando il funzionamento errato guardando contemporaneamente sia le uscite dei tre flip-flop di stato che la linea $\overline{CLR}$ su di un oscilloscopio a 4 canali. Se si osserva uno 0 momentaneo ed errato sulla linea $\overline{CLR}$ e il ritardo nella transizione di FF2, si è in presenza di un problema di temporizzazione piuttosto che di un problema di circuito aperto. Quindi il riparatore dovrà scoprire se il rallentamento del flip-flop è dovuto ad un guasto nel flip-flop stesso o ad un eccessivo carico sui suoi circuiti di ingresso e di uscita. Un eccesso di carico potrebbe essere dovuto ad una dispersione di corrente o ad una capacità a massa che non dovrebbero esistere.

Errata transizione dallo stato 1 allo stato 4

Un altro esempio di un problema di tempificazione che potrebbe verificarsi durante una transizione tra due stati è la transizione dallo stato 1 allo stato 4 quando S è al livello 1. In questo caso entrambi i flip-flop FF3 ed FF1 di *figura 5.6* cambiano contemporaneamente.

Se FF3 è significativamente più veloce di FF1, la macchina potrebbe passare per un istante nella condizione di stato 0 prima di raggiungere lo stato 4. Lo stato 0 momentaneo avrebbe lo stesso effetto di un azzeramento sul totale delle monete inserite e sui flip-flop di selezione dell'articolo; perciò questo provocherebbe l'azzeramento dei flip-flop e la transizione del segnale S allo stato 0. Quindi la macchina si comporterebbe come se non fossero state inserite monete e non ci fosse stata la selezione dell'articolo. Il valore dell'articolo risulterebbe uguale al valore del totale delle monete inserite: perciò non verrebbe restituita nessuna moneta.

Inoltre, quando è generato il segnale VEND, con tutti questi flip-flop di selezione dell'articolo azzerati, non viene generato alcun segnale di rilascio e non viene consegnato alcun articolo all'acquirente.

Se FF1 è significativamente più veloce di FF3 lo stato potrà passare brevemente per lo stato 5 prima di raggiungere finalmente lo stato 4. Sebbene questo segnale resti con tutta probabilità presente per un intervallo di tempo troppo breve, potrebbe comunque essere generato il segnale CHANGE.

Altre possibili transizioni di stato errate

La transizione dallo stato 4 allo stato 5 non ha problemi di temporizzazione potenziali in quanto solo FF3 in *figura 5.6* passa da 0 ad 1. Comunque le transizioni dallo stato 5 allo stato 6, dallo stato 6 allo stato 0, e dallo stato 5 allo stato 0 possono causare problemi di temporizzazione in quanto, in ciascuno di questi casi, due flip-flop di stato cambiano contemporaneamente. Nella transizione dallo stato 5 allo stato 6

Q_0 e Q_1 di FF3 ed FF2 rispettivamente cambiano contemporaneamente. Se FF3 è più veloce di FF2, si avrà uno stato 4 per qualche istante prima dello stato 6. Probabilmente questo non creerà problemi in quanto il segnale VEND generato sarà troppo breve per provocare la consegna di un altro articolo. Se FF2 cambia prima di FF3 si avrà uno stato 7 momentaneo prima dello stato 6. Poiché allo stato 7 non è associata alcuna attività, questo fatto non creerà problemi.

Nel passaggio dallo stato 6 allo stato 0 si potrà avere sia il passaggio momentaneo per lo stato 4 che quello per lo stato 2. Anche in questo caso una qualsiasi breve uscita di impulsi sulla linea VEND sullo stato 4 non dovrebbe avere conseguenze, così come non dovrebbe averne il passaggio per lo stato 2.

Nella transizione dallo stato 5 allo stato 0 si potrà avere un passaggio momentaneo per lo stato 4 o per lo stato 1. Nessuno di questi dovrebbe avere effetti significativi sul funzionamento della macchina. Perciò gli unici problemi di temporizzazione critici nel distributore automatico sarebbero la transizione dallo stato 1 allo stato 2 e la transizione dallo stato 1 allo stato 4. In entrambi i casi potrebbe verificarsi un breve impulso sulla linea $\overline{\text{CLR}}$, il che potrebbe provocare un errore del sistema. Ciascuna di queste condizioni può essere verificata velocemente con un oscilloscopio a 4 canali.

Molti problemi di temporizzazione potrebbero essere evitati coinvolgendo un solo flip-flop in una transizione di stato o utilizzando una seconda serie di flip-flop come generatore per tutti i segnali di controllo e di uscita del sistema.

Considerazioni sul progetto

Tutti questi problemi di temporizzazione avrebbero potuto essere evitati se la macchina fosse stata progettata più accuratamente. Nessuno di questi problemi di temporizzazione si sarebbe verificato se si fosse fatto in modo che tutte le transizioni da uno stato ad un altro coinvolgessero il cambiamento di un solo flip-flop. Avrebbe potuto servire ad eliminare i problemi anche aggiungere un'altra serie di flip-flop di stato. Utilizzando questa seconda serie di flip-flop per generare tutti i segnali di controllo del sistema e di uscita, i momentanei cambi di stato che si verificano in FF1, FF2 ed FF3 non avrebbero avuto alcun effetto sul funzionamento della macchina. Comunque è possibile che anche la migliore macchina sequenziale sincrona presenti ancora alcuni problemi legati alla temporizzazione. Ciò è vero anche per sistemi sequenziali basati su microcomputer. Consideriamo ora alcuni di questi.

PROBLEMI DI TEMPORIZZAZIONE NEI COMPUTER

Nei sistemi di computer le relazioni di temporizzazione sono tutte basate su di un clock. Tutti gli eventi devono verificarsi rispettando certe relazioni temporali.

Un computer è un sistema digitale sequenziale, ma la sua struttura tende ad isolare i problemi di temporizzazione ad un suo particolare sottosistema. Esso è anche un sistema sincrono, così che tutti i segnali di temporizzazione sono collegati tramite relazioni temporali note con il clock o i clock del sistema. Naturalmente queste relazioni devono essere mantenute inalterate se si vuole che il sistema funzioni correttamente. Un'altra caratteristica dei computer è il gran numero di segnali binari che si verificano contemporaneamente (in parallelo). Per osservare in modo corretto le relazioni temporali tra questi segnali sono essenziali oscilloscopi a più canali e spesso sono necessari alcuni speciali strumenti chiamati analizzatori logici per analizzare completamente il funzionamento di un computer. Il funzionamento e l'applicazione degli analizzatori logici sarà trattato nel capitolo seguente.

Problemi di temporizzazione di ingresso/uscita

I dispositivi di ingresso e di uscita più semplici che vengono utilizzati in un computer sono i registri e i flip-flop. I circuiti generano un'uscita dati affidabile quando vengono forniti dal processore i corretti segnali di ingresso e di temporizzazione. Le relazioni temporali descritte in *figura 8.5* si possono applicare a questi tipi di dispositivi di ingresso e uscita.

In un microcomputer i dati per un'operazione di uscita sono presenti per un intervallo di tempo breve rispetto al segnale di clock. Analogamente i dati per un'operazione di ingresso devono essere a disposizione del processore in un preciso istante.

Le *figure 7.1* e *7.2* sono esempi di semplici circuiti di ingresso e di uscita. Questi circuiti di ingresso e di uscita a mappa di bit sono rappresentativi di molti circuiti sincroni.

La temporizzazione fondamentale del circuito di uscita è mostrata in *figura 8.10*. I dati vengono emessi sulla linea CRUOUT nello stesso istante in cui i segnali di indirizzo vengono emessi dal processore. Queste uscite sono mostrate in *figura 8.10* come livelli 1. Un po' più tardi in questo ciclo viene inviato dal processore il segnale CRUCLK. Solitamente il dato viene acquisito in corrispondenza del fronte di discesa dell'impulso di clock. Il dato e gli indirizzi devono essere presenti per circa 500 nanosecondi dall'istante in cui si è verificato questo fronte di discesa. Questo tempo è molto maggiore dei 20 nanosecondi che approssimativamente costituiscono il normale tempo di stabilizzazione di registri e flip-flop. Il dato e gli indirizzi rimangono poi sui componenti di uscita per circa 160 nanosecondi dopo che si è verificato il fronte di discesa del clock. Anche questo tempo è molto più lungo del tipico tempo di mantenimento di 5 nanosecondi richiesto per registri e flip-flop.

Figura 8.10. Temporizzazione d'uscita a mappa di bit della famiglia 9900

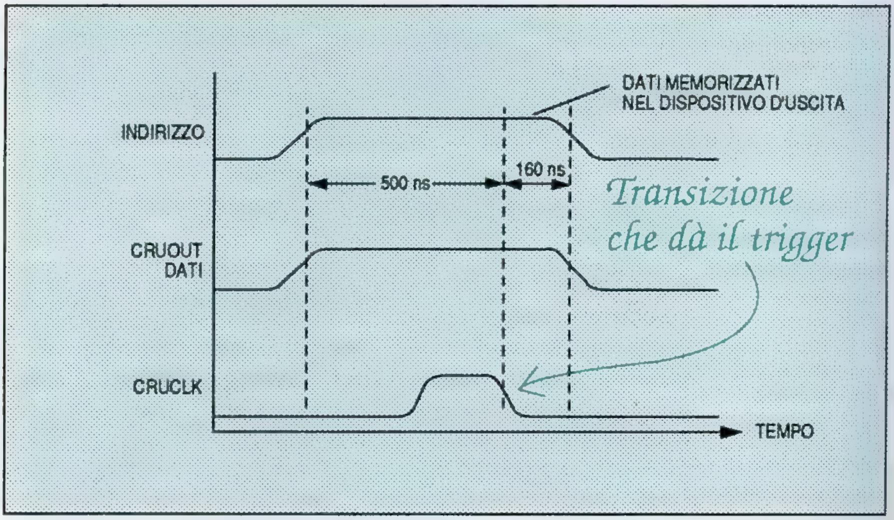
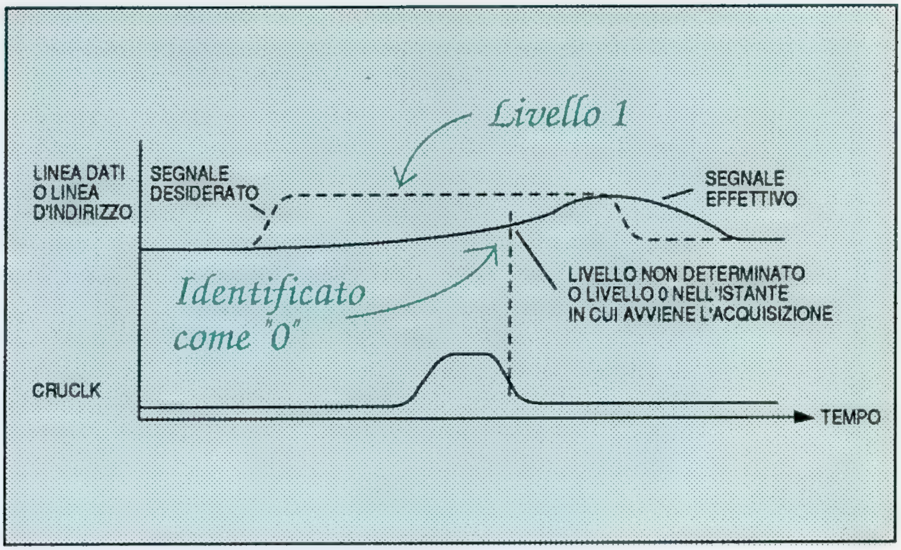


Figura 8.11. Guasto sulla linea d'indirizzo o sulla linea dati di un trasferimento I/O a mappa di bit


Livello 1
LINEA DATI O LINEA D'INDIRIZZO
SEGNALE DESIDERATO
SEGNALE EFFETTIVO
LIVELLO NON DETERMINATO O LIVELLO 0 NELL'ISTANTE IN CUI AVVIENE L'ACQUISIZIONE
Identificato come "0"
CRUCLK
TEMPO

Segnali ritardati

I ritardi nei segnali sono particolarmente preoccupanti. Se il segnale è un indirizzo sarà trasferito un bit sbagliato. Se il segnale è un dato sarà trasferito un livello sbagliato.

Anche con questi margini di tempo relativamente generosi potrebbero esserci problemi di temporizzazione. Uno dei possibili è mostrato per esempio in *figura 8.11*. Per qualche motivo il segnale dei dati oppure uno o più segnali di indirizzo sono in ritardo di 500 nanosecondi o più. A causa di questo ritardo il segnale non ha raggiunto il livello 1 nell'istante in cui si verifica il fronte di discesa del segnale CRUCLK. Se il segnale in ritardo è un bit di indirizzo, ci sarà sulle linee un indirizzo errato e verrà trasferito il bit sbagliato. Se il segnale in ritardo è un bit di uscita, verrà emesso uno 0 al posto di un 1. Ciò produrrà naturalmente risultati errati. Per esempio un dispositivo esterno che dovesse essere abilitato da un bit 1 rimarrebbe invece disabilitato. Se il segnale in ritardo è un bit di ingresso, il processore inizierà la sua azione successiva sulla base di informazioni sbagliate.

Un oscilloscopio a più canali aiuta nel localizzare i problemi della temporizzazione. Esso mostrerà il tempo di salita eccessivamente ritardato del segnale d'indirizzo o del segnale di dati.

Questo problema può essere individuato osservando contemporaneamente i segnali dei dati, degli indirizzi e il segnale CRUCLK su di un oscilloscopio a più canali in modo da notare la crescita troppo lenta (tempo di salita troppo lungo) di uno di questi. Se è possibile inserire un'istruzione di prova nel processore, si può scrivere una sequenza di LOOP che fa cambiare continuamente tra 0 e 1 il bit di I/O scelto.

Una volta entrata in funzionamento questa sequenza di LOOP, essa indirizzerebbe continuamente un bit di I/O prescelto cosicché si potrebbero esaminare i segnali di indirizzo e i segnali dei dati.

Se viene selezionato il bit di I/O sbagliato, allora il problema si verifica in una linea di indirizzo. Si supponga, per esempio, che l'indirizzo di base per le operazioni di uscita sia 080H ma che venga selezionata l'uscita all'indirizzo 000H mentre l'uscita all'indirizzo 080H non venga mai trasferita alle linee di uscita. Ciò significa che il bit che si supponeva essere ad 1 nel codice di indirizzo (l'ottavo bit di indirizzo) rimane invece a 0. Per questo il codice di indirizzo desiderato 0000 $\underline{1}$000 0000 viene sempre sostituito da 0000 0000 0000. Con questa informazione la linea di indirizzo di cui sospettare si trova facilmente utilizzando l'oscilloscopio.

Se un segnale appare effettivamente sulla giusta linea di uscita, le linee di indirizzo e il segnale CRUCLK funzionano correttamente. Comunque deve esserci un problema nel segnale dei dati, la cui temporizzazione andrebbe verificata rispetto al segnale CRUCLK e alle linee di indirizzo. Un'attenta osservazione e un'accurata misurazione possono rivelare che, sebbene ci sia in uscita un segnale, questo è , in realtà, il complementare di quello che dovrebbe essere. Ciò significa che l'uscita è 0 quando dovrebbe essere 1 e viceversa. Questo indicherebbe un problema nella linea dati, che può essere esaminata per individuare il problema fisico del circuito. Naturalmente, se l'uscita è bloccata a 0 o ad 1, il problema può essere messo in relazione con la temporizzazione solo se il segnale CRUCLK è stato ritardato o si è in qualche modo deteriorato. Se il segnale CRUCLK è normale, allora la condizione di blocco a 0 o ad 1 è causata da un guasto nel circuito o nei fili del tipo descritto nei precedenti capitoli.

Problemi di temporizzazione delle memorie

Il fatto che un dispositivo di memoria sia o meno in grado di rispondere abbastanza velocemente da soddisfare i requisiti di temporizzazione di un processore dipende dalle sue caratteristiche di temporizzazione.

Temporizzazione di scrittura

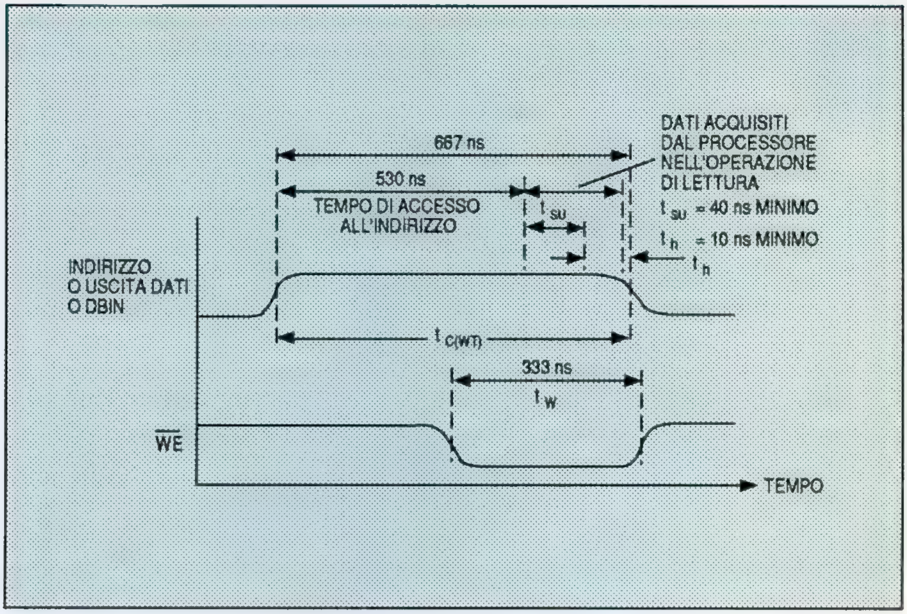
Le caratteristiche dei segnali di controllo della memoria per una tipica operazione di memoria di un processore della famiglia 9900 sono mostrate in *figura 8.12*. La sequenza delle operazioni è la seguente: i dati e gli indirizzi vengono inviati al sottosistema di memoria all'inizio del ciclo e il segnale di controllo di scrittura $\overline{WE}$ viene inviato subito dopo. I segnali dei dati e i segnali di indirizzo sono presenti per un totale di circa 667 nanosecondi, mentre $\overline{WE}$ è al suo livello attivo basso per circa 333 nanosecondi.

Il funzionamento corretto della memoria dipende dalle esatte relazioni temporali nell'applicazione dei segnali d'indirizzo, dei segnali di abilitazione alla scrittura e alla lettura e dei segnali di dati.

Perché la scrittura avvenga esattamente, il decodificatore di memoria deve avere selezionato la locazione di memoria indirizzata. L'indirizzo e il dato devono essere disponibili per un tempo minimo, $t_{c(wr)}$, detto ciclo di scrittura. Una volta che la locazione di memoria è indirizzata, il dato deve essere presente abbastanza a lungo per poter essere memorizzato dopo la transizione di $\overline{WE}$ al livello basso. La cosa più importante è che l'indirizzo e il dato devono essere disponibili nel periodo di tempo t_w, nel quale è presente il segnale di abilitazione di scrittura $\overline{WE}$. In questi circuiti integrati di memoria il dato non viene memorizzato nel dispositivo in corrispondenza del fronte di salita del segnale di abilitazione di scrittura, ma è conservato in un dispositivo di memorizzazione asincrono come un flip-flop R-S oppure, nel caso di una memoria dinamica, in un condensatore. L'operazione di scrittura non avrà un esito positivo se:

1. I segnali di abilitazione di scrittura e di indirizzo non sono presenti per un tempo abbastanza lungo quando i dati sono disponibili.

Figura 8.12. Temporizzazione di base di una memoria in un microprocessore della famiglia 9900


667 ns
530 ns
TEMPO DI ACCESSO ALL'INDIRIZZO
t_{su}
DATI ACQUISITI DAL PROCESSORE NELL'OPERAZIONE DI LETTURA
t_{su} = 40 ns MINIMO
t_h = 10 ns MINIMO
t_h
INDIRIZZO O USCITA DATI O DBIN
$t_{C(WT)}$
333 ns
t_W
$\overline{WE}$
TEMPO

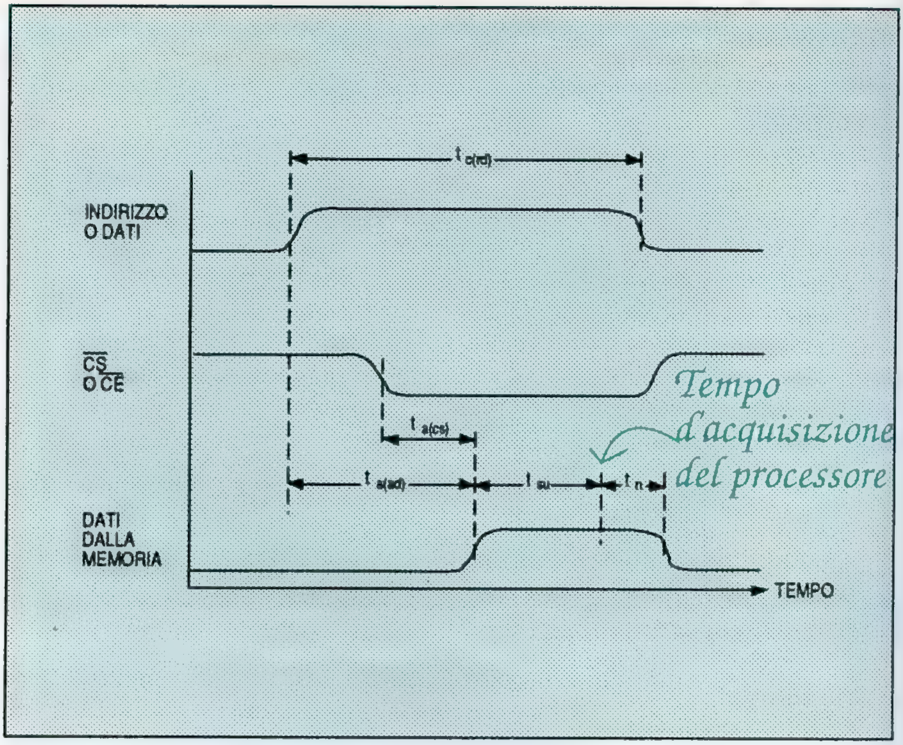
2. L'impulso di abilitazione di scrittura è in ritardo di così tanto tempo che i segnali di indirizzo e i segnali dei dati vengono rimossi prima che si verifichi l'abilitazione alla scrittura.
3. Il segnale di indirizzo è ritardato così tanto che la locazione desiderata non può essere selezionata dal dispositivo nell'intervallo di tempo.
4. Il segnale dati è ritardato così tanto che i dati non sono presenti abbastanza a lungo.

Temporizzazione di lettura

Il funzionamento corretto della lettura dipende dal fatto che la memoria sia in grado di mantenere i dati sul bus per un tempo abbastanza lungo da permettere ai dati stessi di essere memorizzati nell'appropriato registro di memorizzazione del processore.

La temporizzazione di lettura di memoria è più semplice perché il segnale di indirizzo e il segnale di controllo di lettura della memoria (DBIN) vengono inviati contemporaneamente o quasi. Essi arrivano alla memoria all'inizio di un ciclo di lettura ed entrambi rimangono presenti per tutti i 667 nanosecondi. Perché il processore riceva i dati la memoria deve mantenerli sul bus dati per fornire un tempo di stabilizzazione (t_{su}) e di mantenimento (t_h) sufficienti. Ciò permetterà al processore di memorizzare i dati in un registro interno finché non sarà pronto per utilizzarli. Se non vengono rispettati i requisiti del processore di un tempo di stabilizzazione di 40 nanosecondi e di un tempo di mantenimento di 10 nanosecondi si avrà un errore nei dati. La temporizzazione del segnale per un'operazione di lettura in un tipico sottosistema di memoria è mostrata in *figura 8.13*. Il minimo tempo per un intero ciclo di lettura, $t_{c(rd)}$, viene specificato per ciascun tipo di circuito integrato di memoria insieme ai suoi tempi di accesso. La notazione $t_{a(ad)}$ indica il tempo impiegato dalla memoria per emettere un dato sul bus dati dopo che questo è stato indirizzato dal pro-

Figura 8.13. Temporizzazione tipica della lettura di memoria



cessore. Analogamente il tempo di accesso $t_{a(cs)}$ è il tempo che intercorre fra l'arrivo del segnale di selezione del chip ($\overline{CS}$) o di abilitazione dell'uscita ($\overline{OE}$) e il momento in cui i dati sono posti sulle linee. I dati devono restare sul bus dati per un certo tempo prima che il processore li acquisisca nel suo registro dati. Questo è il tempo di stabilizzazione che viene indicato come $t_{(su)}$ in *figura 8.13*. Analogamente i dati devono rimanere stabili sul bus dati per un brevissimo intervallo di tempo dopo che il processore genera un segnale di acquisizione dati. Questo tempo di mantenimento, indicato con t_h, lascia ai flip-flop del registro dati il tempo necessario per memorizzare i dati stessi.

In un progetto affidabile il massimo tempo di accesso all'indirizzo di un processore dovrebbe superare il tempo di accesso alla memoria, in modo tale da fornire un prestabilito margine di tempo per il deterioramento dei componenti.

Cause dei ritardi di temporizzazione

Alcuni problemi di temporizzazione sono tipici delle operazioni di lettura o di scrittura.

Si faccia riferimento alla *figura 8.12*. Il tempo massimo di accesso all'indirizzo ($t_{a(ad)}$) che può essere accettato per questo tipo di processore è di 530 nanosecondi. Se viene selezionato un dispositivo di memoria il cui tempo di accesso all'indirizzo è di 500 nanosecondi, si avrà un margine di sicurezza di 30 nanosecondi per l'invecchiamento del componente. Se si verifica un guasto in una delle linee di indirizzo o di selezione del chip tale che questo margine di sicurezza interno venga superato, l'operazione di lettura della memoria non produrrà dati o trasferimenti di istruzioni al processore affidabili. Il problema può essere individuato solo attraverso misurazioni molto accurate delle forme d'onda degli indirizzi e dei dati presenti sui piedini del circuito di memoria. Tali misurazioni dovrebbero verificare che siano soddisfatti i tempi di accesso specificati nei manuali forniti dai costruttori.

In alcuni sistemi una memoria relativamente lenta potrebbe essere stata accoppiata con un veloce processore, cosicché il tempo in cui è attivo il segnale di abilitazione di scrittura è appena sufficiente per il dispositivo di memoria. Quindi, poiché i componenti si deteriorano con il passare del tempo e, di conseguenza, aumentano i ritardi, possono verificarsi degli errori durante le operazioni di scrittura.

Alcuni problemi di memoria e di temporizzazione sono gli stessi sia che si tratti di un'operazione di lettura sia che si tratti di un'operazione di scrittura.

È possibile che si presentino gli stessi problemi di temporizzazione sia che si stia svolgendo un'operazione di lettura in un sottosistema di memoria sia che se ne stia svolgendo una di scrittura. Le principali cause dei problemi di temporizzazione sono riassunte in *figura 8.14*, mentre i componenti che possono essere fonte di problemi sono mostrati in *figura 8.14a*. I circuiti vengono semplificati rappresentandoli nella forma più generale di circuiti equivalenti. Il generatore di corrente, I_D, rappresenta il circuito che costringe la tensione del segnale (V_C) ad essere 0 od 1. L'uscita del circuito del generatore di corrente è un livello logico 0 od 1. Se si tratta di un'operazione di lettura in memoria, esso potrebbe essere un circuito all'interno della memoria stessa; se fosse invece un'operazione di scrittura in memoria esso potrebbe essere un circuito interno al processore. Il resistore e il condensatore, disegnati nel quadrato tratteggiato, rappresentano il circuito di carico che riceve il segnale. Questo circuito di carico può provocare dei cambiamenti nella forma del segnale di uscita.

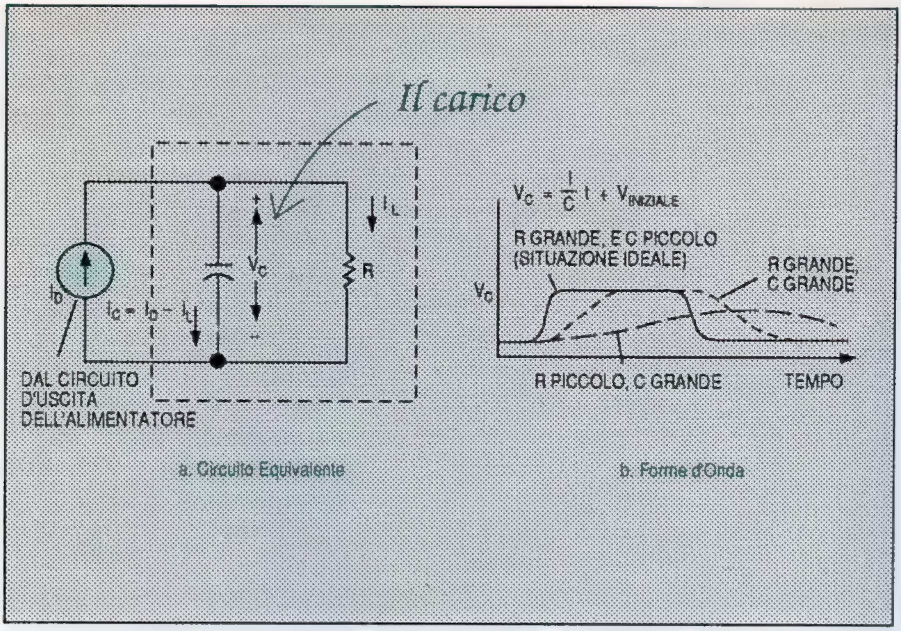
Il circuito di uscita deve caricare o scaricare il condensatore (C) mantenendo allo stesso tempo il flusso di corrente nel resistore. La corrente (I_C) disponibile per caricare o scaricare il condensatore è la corrente del circuito di uscita, I_D, meno la corrente nel resistore, I_L ($I_C = I_D - I_L$). La corrente I_C fa sì che la tensione V_C salga al livello 1 o scenda al livello 0 con una rapidità proporzionale al valore di I_C e inversamente proporzionale al valore della capacità.

In ogni istante, anche per valori costanti di I_D, il valore di I_C può variare. Ciò avviene perché il valore di I_L è proporzionale a V_C e significa che se V_C cambia, anche I_L cambia. In un tipico circuito che riceve il segnale di uscita la resistenza è molto elevata, cosicché I_L assume sempre valori molto piccoli. In questo caso I_C viene mantenuto a un valore molto vicino ad I_D, I_D può variare a seconda della natura del circuito di uscita e, di conseguenza, I_C varierà con essa. Anche con un I_D variabile il condensatore si caricherà ad un livello 1 o si scaricherà ad un livello 0 in un tempo relativamente veloce. Il risultato sarebbe la forma d'onda rappresentata con una linea continua in *figura 8.14b* (R grande, C piccolo). I tempi di salita, di discesa e di ritardo sono molto brevi e non dovrebbero provocare dei problemi in un circuito ben progettato.

Le capacità distribuite nel circuito dovute ai componenti, ai fili conduttori o alla lunghezza possono influenzare il tempo di salita e di discesa e causare i ritardi degli impulsi.

Quando il valore della capacità aumenta significativamente a causa di un problema di disposizione sul circuito stampato o di cablaggio, V_C impiega molto più tempo per raggiungere il livello 0 o il livello 1 desiderato. Il risultato sarebbe la forma d'onda punteggiata mostrata in *figura 8.14b* (R grande, C grande). In questo caso i tempi di salita, di discesa e di ritardo sono lunghi, il che può causare errori nelle operazioni di lettura o nelle operazioni di scrittura o in entrambe. Ciò dipenderà dal fatto che l'e-

Figura 8.14. Effetto dei guasti in circuiti collegati con un driver del bus


Il carico
I_D
$I_C = I_D - I_L$
I_L
V_C
R
+
-
DAL CIRCUITO D'USCITA DELL'ALIMENTATORE
$V_C = \frac{I}{C} t + V_{INIZIALE}$
R GRANDE, E C PICCOLO (SITUAZIONE IDEALE)
R GRANDE, C GRANDE
V_C
R PICCOLO, C GRANDE
TEMPO

a. Circuito Equivalente

b. Forme d'Onda

levata capacità si trovi su una linea di indirizzo, su una linea di dati o su una linea di controllo.

Solitamente la capacità su di una linea non cambia in maniera significativa col tempo e quindi il problema precedentemente descritto non si verifica in sistemi progettati correttamente. Se un sistema, dopo che è stato progettato, risulta avere un margine di sicurezza molto piccolo sui tempi di accesso, lettura e scrittura in memoria, allora anche i cambiamenti molto piccoli nelle capacità di una linea o di un componente potrebbero provocare problemi. Il guasto può essere individuato abbastanza facilmente esaminando le forme d'onda delle linee di dati, di indirizzo e di controllo e misurando le relazioni che intercorrono tra la temporizzazione d'accesso e le altre.

In un circuito in cui il carico è rappresentato da un resistore in parallelo con un condensatore, un cammino a bassa resistenza posto in parallelo con il resistore ridurrà il tempo di salita della tensione ai capi del carico in maniera significativa.

Il problema più comune e più grave che si può presentare è che si crei un percorso a bassa resistenza ai capi del condensatore. Ciò può essere causato da variazioni nel circuito di ingresso che riceve il segnale o ad una particella conduttrice tra i conduttori del circuito. Questa particella conduttrice (da non confondersi con un corto circuito) fornirebbe un percorso di dispersione della corrente attorno al resistore. In entrambi i casi la corrente disponibile per caricare il condensatore è significativamente ridotta. Ciò aumenta di molto il tempo richiesto per caricare il condensatore, cioè per portare la sua tensione al livello 1. Infatti, esso non potrebbe nemmeno raggiungere il livello 1 prima che I_D cambi dando inizio alla scarica del condensatore che lo porterà al livello 0. Questo effetto è mostrato dalla forma d'onda rappresentata con una linea tratteggiata in *figura 8.14b* (R piccolo, C grande).

Effetti dei ritardi di temporizzazione

Vediamo ora che cosa succede se una tale forma d'onda (R piccolo, C grande) appare sulla linea dei dati, sulla linea di indirizzo o sulla linea di controllo. Se un tale segnale appare sulla linea di indirizzo, i dati verranno letti dalla (o scritti nella) locazione sbagliata perché il codice di indirizzo non verrebbe riconosciuto correttamente. Se un tale segnale appare su una delle linee dati, si accederebbe alla locazione corretta, ma non sarebbe possibile scrivere in maniera affidabile un 1 nella posizione relativa a quel bit.

Se il problema si verificasse su una linea di controllo che abilita alla lettura o alla scrittura, non sarebbe possibile nè leggere nè scrivere (a seconda della linea di controllo su cui si verifica il ritardo) nei dispositivi del sistema. Anche se il segnale raggiungesse un 1 logico in un certo istante, non raggiungerebbe questo livello in tempo per fornire operazioni di lettura e di scrittura corrette. Se, per esempio, il problema si fosse verificato su di una linea di abilitazione alla scrittura, il sottosistema di memoria potrebbe rimanere in una condizione di scrittura permanente. Se ciò invece si verificasse per una linea di abilitazione alla lettura, il sottosistema di memoria potrebbe essere posto in una condizione di lettura permanente.

Controllando un'operazione di lettura o di scrittura ripetitiva nota, l'oscilloscopio dovrebbe mostrare se una linea è lenta nel cambiare i livelli.

Le forme d'onda potrebbero essere osservate facilmente con un oscilloscopio che faccia apparire sul suo schermo le linee di indirizzo, le linee dati e le linee di controllo durante un'operazione di lettura o di scrittura ripetitiva e nota. Se, nel corso di tale misurazione, uno dei segnali mostrasse una transizione molto lenta dal livello 0 al livello 1, o viceversa, si dovrebbe sospettare il malfunzionamento di quella linea. Se, di tanto in tanto, una linea dati o una linea di indirizzo indicasse un 1 quando dovrebbe invece indicare 0, o viceversa, mentre in altri istanti il valore indicato fosse corretto, si dovrebbe pensare ad un guasto intermittente. I guasti intermittenti sono più difficili da individuare. Il riparatore deve usare ogni risorsa per determinare se la causa del guasto intermittente è un corto circuito o un circuito aperto. Con un buon uso dell'oscilloscopio ed un esame accurato di tutti i collegamenti e componenti nell'area sospetta, il guasto può essere individuato.

Ricerca delle cause dei ritardi di temporizzazione

Una tecnica per trovare la causa dei problemi dovuti ai ritardi di temporizzazione è di isolare tutti i carichi dalla linea che presenta il guasto e osservare il segnale della linea. Se si osserva ancora la condizione errata, il guasto sta nel dispositivo che genera il segnale o nel filo che fornisce il collegamento del segnale con gli altri componenti.

Le tecniche di immissione dei segnali d'impulso aiuteranno nel determinare se il deterioramento della forma d'onda è dovuto ad un componente o ai collegamenti tra i componenti.

Una possibilità è di isolare la linea che porta il segnale di uscita e guidarla con un'onda quadra proveniente da un generatore di segnali: si tratta quindi di osservare la forma d'onda. Viene utilizzato cioè un generatore di segnali che fornisce un segnale noto, la frequenza, l'ampiezza dell'impulso e il periodo nel quale possono essere programmati dall'utente. Se la forma d'onda presenta ancora dei tempi di salita e di discesa rapidi, allora il dispositivo che fornisce il segnale di uscita deve essere guasto. Se la forma d'onda presenta tempi di salita e di discesa relativamente lunghi, allora deve esserci una resistenza bassa verso massa sulla linea oppure una capacità inaspettatamente grande verso massa sulla linea. Una volta noti i sintomi del guasto, un'ispezione visiva potrebbe essere il modo più veloce per localizzare la causa del guasto stesso. Un'altra tecnica potrebbe essere quella di isolare alcune parti scelte della linea e osservare la forma d'onda in corrispondenza di differenti locazioni lungo la linea.

Se il segnale d'uscita del dispositivo è corretto quando tutti i carichi vengono rimossi dalla linea, uno degli ingressi deve essere difettoso. Ricollegando gli ingressi uno alla volta finché viene osservato nuovamente il deterioramento del segnale, può essere trovato il circuito guasto.

Questo procedimento funziona indipendentemente dal fatto che il guasto si trovi su una linea dati, su una linea di indirizzo o su una linea di controllo. Se la sorgente del segnale errato è un ingresso di lettura in memoria o di I/O, si deve isolare per primo il circuito di uscita della memoria o quello di I/O. In effetti, se la scrittura in memoria funziona in maniera soddisfacente, la linea dati stessa dovrebbe essere in buone condizioni. Perciò se l'ingresso di lettura in memoria o di I/O non funziona correttamente la causa più probabile del problema dovrebbe essere il circuito di uscita della memoria, o il sottosistema che genera la lettura I/O (per esempio una tastiera o un registratore lettore di cassette), oppure il circuito di ingresso del processore.

CHE COSA ABBIAMO IMPARATO?

1. Molte delle tecniche usate per isolare i guasti relativi a problemi di livello in linee di segnali costanti o lentamente variabili possono essere impiegate anche per localizzare i guasti in linee di segnali rapidamente variabili o i guasti che sono in relazione con la temporizzazione o la forma dei segnali digitali.
2. Per misurare le caratteristiche dei livelli di tensione di segnali rapidamente variabili o per misurare le caratteristiche temporali deve essere utilizzato l'oscilloscopio invece dei semplici voltmetri e delle sonde logiche.
3. Il funzionamento non corretto di un sistema o di un sottosistema spesso può essere attribuito ad un segnale che arriva troppo tardi.
4. Un segnale può arrivare tardi a causa di: (a) un'insufficiente corrente di pilotaggio proveniente dal circuito che genera il segnale, (b) un carico eccessivo sulla linea dovuto a guasti sulle linee o sui circuiti di ingresso, (c) dispositivi che hanno un eccessivo ritardo di propagazione nei loro circuiti di uscita.

Quiz del Capitolo 8

1. Il tempo di salita di un segnale è il tempo necessario:
   a. Per compiere un transizione da un livello basso a un livello alto.
   b. Per compiere una transizione da un livello alto a un livello basso.
   c. Perché il segnale di uscita inizi a compiere una transizione verso un livello alto a partire dal momento in cui un segnale di ingresso provoca la transizione.
   d. Tra i punti medi di un segnale di onda quadra.

2. I tempi di salita e di discesa vengono misurati:
   a. Tra i punti al 50% su una forma d'onda.
   b. Tra l'istante in cui il livello di tensione è 0 e quello in cui il livello di tensione è 1.
   c. Tra i punti al 10% e al 90% nella transizione di tensione.
   d. Tra il segnale di ingresso e quello di uscita.

3. Il tempo di mantenimento per un flip-flop di tipo D è il tempo in cui:
   a. I dati vengono mantenuti sull'uscita dopo il segnale di clock.
   b. I dati devono essere tenuti all'ingresso dopo un fronte di clock.
   c. Il clock deve essere alto.
   d. Il clock deve essere basso.
   e. I dati devono essere stabili prima di un fronte di clock.

4. Il tempo di stabilizzazione per un flip-flop di tipo D è il tempo in cui:
   a. I dati vengono mantenuti sull'uscita dopo il segnale di clock.
   b. I dati devono essere tenuti all'ingresso dopo un fronte di clock.
   c. Il clock deve essere alto.
   d. Il clock deve essere basso.
   e. I dati devono essere stabili prima di un fronte di clock.

5. L'ampiezza di un impulso di clock attivo alto è il tempo in cui:
   a. I dati vengono mantenuti sull'uscita dopo il segnale di clock.
   b. I dati devono essere tenuti all'ingresso dopo un fronte di clock.
   c. Il clock deve essere alto.
   d. Il clock deve essere basso.
   e. I dati devono essere stabili prima di un fronte di clock.

6. I periodi di clock sono misurati:
   a. Tra i punti corrispondenti sulla forma d'onda del clock.
   b. Dal livello basso al livello alto.
   c. Dal livello alto al livello basso.
   d. Dall'istante in cui l'impulso di clock è al 50% della sua transizione dal livello basso al livello alto fino all'istante in cui esso si trova al 50% della sua transizione dal livello alto al livello basso.

7. Se la frequenza di clock è di 1000 hertz, quant'è il suo periodo in millisecondi?
   a. 0,001
   b. 1
   c. 1000
   d. 10000

8. Che cosa mostrerà sullo schermo un oscilloscopio con l'ingresso di segnale a massa e lo sweep sicronizzato internamente e automaticamente?
   a. Un punto.
   b. Una linea verticale.
   c. Una linea orizzontale.
   d. Uno schermo vuoto.

9. Se un dispositivo di memoria ha un tempo di accesso all'indirizzo di 500 nanosecondi, quale sarà, in nanosecondi, il tempo minimo per un ciclo di memoria necessario per utilizzare il dispositivo con un processore che ha un tempo di stabilizzazione di 100 nanosecondi e un tempo di mantenimento di 25 nanosecondi?
   a. 500
   b. 600
   c. 625
   d. 750

10. Se il tempo di salita della tensione della linea è di 20 nanosecondi quando una capacità di linea di 150 pF viene caricata con una corrente costante ID, quale sarà approssimativamente il tempo di salita, in nanosecondi, se la capacità della linea viene aumentata a 300 pF?
   a. 10
   b. 20
   c. 30
   d. 40

TECNICHE AVANZATE

SCOPO DEL CAPITOLO

Nella maggior parte delle tecniche studiate finora le relazioni temporali non sono state critiche per le operazioni del sistema. All'aumentare della velocità operativa del sistema le relazioni temporali tra i segnali diventano più critiche e si rende necessario l'uso di tecniche di misurazione avanzate che utilizzano oscilloscopi a più canali o altri strumenti a più canali e ad alta velocità per verificare se vengono mantenute le giuste relazioni temporali. Inoltre, poiché i circuiti integrati nei sistemi divengono più complessi, si rendono necessarie tecniche differenti. Con l'aumentare del numero delle locazioni di memorizzazione oppure del numero e dei tipi di funzioni di I/O, le tecniche di ricerca ed eliminazione dei guasti devono diventare più sofisticate e, in molti casi, totalmente automatiche. In questo capitolo verranno trattate alcune delle tecniche più avanzate che devono essere utilizzate in tali sistemi digitali complessi.

PROBLEMI DELLE MEMORIE A GRANDE SCALA

Per il test di memorie su larga scala, se si utilizzano solo le tecniche più semplici non si dispone di un tempo adeguato per svolgere una completa sequenza di autodiagnosi.

Quando la grandezza della memoria di un sistema supera le 64.000 locazioni, la gamma dei metodi di ricerca dei guasti realmente utilizzabili diviene sempre più limitata. Un approccio, come citato nel capitolo 6, è quello di fornire al sistema un programma di prova che verifichi automaticamente il funzionamento dell'intera memoria allo scopo di determinare gli indirizzi nei quali si verificano gli errori. Una volta localizzato un errore, il programma stamperà un messaggio che guiderà il riparatore nel suo lavoro. Questo metodo presenta però un problema: richiede troppo tempo. Se ci sono milioni di locazioni nella memoria del sistema ed occorrono parecchi microsecondi perché il programma verifichi il funzionamento di ciascuna locazione, il software diagnostico impiegherà parecchi minuti per completare la prova di memoria. Ciò potrebbe essere inaccettabile durante l'accensione o una sequenza di autodiagnosi. Perciò considereremo ora un metodo nel quale l'hardware incorporato prova automaticamente ogni locazione di memoria ogniqualvolta la locazione viene utilizzata durante le operazioni di routine.

Struttura della memoria autodiagnostica

Una tecnica che viene utilizzata nelle memorie di grande capacità è mostrata nelle *figure* dalla *9.1* alla *9.3*. La struttura di memoria, mostrata in *figura 9.1*, è quella di una RAM dinamica che utilizza dispositivi simili a quelli discussi nel capitolo 6. Ciascun dispositivo mostrato in *figura 9.1* memorizza 65536 singoli bit di informazione; cioè un bit per locazione. Utilizzando 16 dispositivi di questo tipo per ogni riga della memoria, possono essere memorizzate 65536 parole di 16 bit ciascuna. (Nel capitolo 6 ciascuna riga era progettata in modo tale da immagazzinare un byte di 8 bit; in *figu-*

ra 9.1 ciascuna riga memorizza 2 byte da 8 bit o una parola da 16 bit.) Anche in *figura 9.1* sono mostrate 16 righe di 16 dispositivi che forniscono una memorizzazione complessiva di 1.024.000 parole di 16 bit utilizzando un totale di 256 circuiti. Questa memoria richiede 20 bit per un indirizzo completo. I 16 bit meno significativi vanno agli ingressi del circuito di memoria compresi tra A_0 e A_7 (8 bit in multiplexing per volta per un totale di 16 bit) e i 4 bit successivi vanno ai quattro ingressi di selezione della riga del decodificatore 4 a 16.

Il decodificatore è abilitato con un'opportuna combinazione delle restanti linee di indirizzo (se ve ne sono) e con un segnale di abilitazione di memoria ($\overline{MEMEN}$) proveniente dal processore. I segnali per la scrittura in memoria ($\overline{W}$) e i segnali di temporizzazione di riga e di colonna ($\overline{RAS}$ e $\overline{CAS}$ rispettivamente) vengono inviati agli ingressi corrispondenti su ciascun circuito di memoria. Infine il segnale di abilitazione di lettura (DBIN o $R/\overline{W}$) viene inviato all'abilitazione di uscita del circuito di correzione dell'errore e il segnale di abilitazione di scrittura ($\overline{WE}$ o $R/\overline{W}$) viene inviato all'abilitazione d'uscita del generatore di parità.

Sono questi circuiti di buffer insieme ai circuiti addizionali che forniscono i segnali da P_1 a P_6 i quali fanno sì che la memoria si autodiagnostichi. I circuiti addizionali sono il circuito del generatore di parità, le locazioni di memoria che immagazzinano i segnali di parità da P_1 a P_6 per ciascuna riga di memorizzazione dei dati e il circuito di correzione degli errori. Questi circuiti permettono la ricerca automatica dell'errore, la sua individuazione e la sua correzione. Sebbene questi circuiti aumentino il costo della memoria del 40% circa, essi garantiscono un funzionamento più affidabile della stessa e diminuiscono i tempi e i costi della ricerca dei guasti.

Generazione di parità

Ulteriori bit di parità vengono aggiunti a ciascuna riga al fine di fornire dati per i circuiti di identificazione e di correzione degli errori, che tentano di rendere autodiagnostico l'esempio di memoria riportato.

Per capire come vengono progettati e come funzionano i generatori di parità e i circuiti di correzione e di individuazione degli errori, devono essere esaminate le informazioni contenute nelle *figura* dalla *9.2* alla *9.4*. L'idea di base è che dato che l'informazione è scritta nelle 16 colonne più a sinistra (D_0 - D_{15}) di una data riga o di un banco di circuiti di memoria, nei 6 circuiti più a destra dello stesso banco o riga di circuiti è scritto un codice di parità (P_1 - P_6).

Il codice di parità viene inserito nei bit di parità (da P_1 a P_6) così da fornire una parità dispari per i bit dati selezionati come mostrato in *figura 9.2*. Poiché viene generata una parità dispari, il bit posto in una particolare locazione di memoria dei bit di parità fa sì che la somma dei bit dei dati che sono 1 e dei bit di parità sia un numero dispari. Perciò il bit di parità P_1, che contiene la parità dei bit dei dati da D_0 a D_8, sarà un 1 se i circuiti di memoria che memorizzano da D_0 a D_8 contengono un numero pari di 1. Se, per esempio, il dato nella locazione a cui si sta accedendo fosse:

$$\text{FOF3H} = 1111\ 000\underline{0}\ \underline{1111}\ \underline{0011}$$

allora i bit sottolineati, quelli compresi tra D_0 e D_8, conterrebbero sei 1, cioè un numero pari di 1. Di conseguenza, il circuito generatore di parità memorizzerà un 1 nel bit P_1 in modo tale da rendere sette gli 1, cioè in modo tale da avere un numero dispari di 1. Il bit di parità P_2 controlla i bit dati D_{15}, D_{13}, D_{11}, D_9, D_8, D_7, D_5, D_3 e D_1

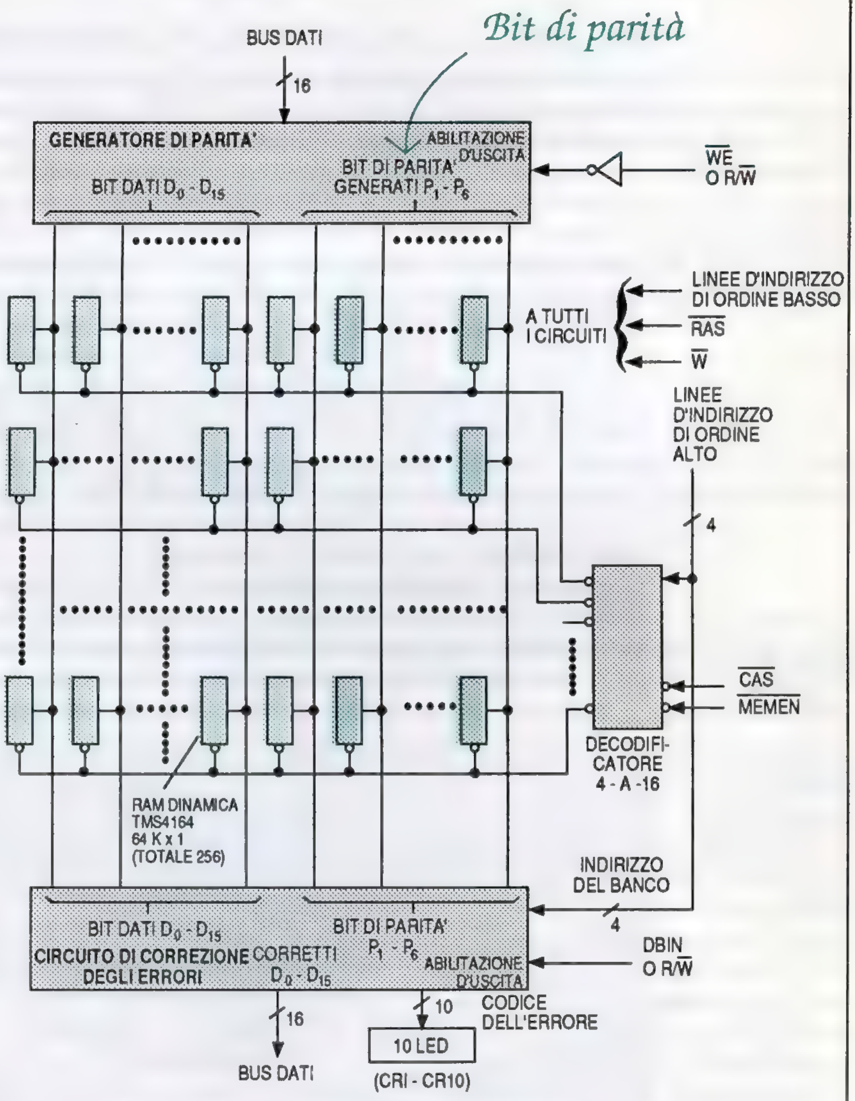


Figura 9.1. Memoria RAM dinamica di grande capacità con individuazione e correzione automatica degli errori

Bit di parità
BUS DATI
16
GENERATORE DI PARITA'
ABILITAZIONE D'USCITA
BIT DI PARITA' GENERATI P_1 - P_6
BIT DATI D_0 - D_{15}
$\overline{WE}$ O R/$\overline{W}$
LINEE D'INDIRIZZO DI ORDINE BASSO
$\overline{RAS}$
$\overline{W}$
A TUTTI I CIRCUITI
LINEE D'INDIRIZZO DI ORDINE ALTO
4
$\overline{CAS}$
$\overline{MEMEN}$
DECODIFICATORE 4 - A -16
RAM DINAMICA TMS4164 64 K x 1 (TOTALE 256)
INDIRIZZO DEL BANCO
4
BIT DATI D_0 - D_{15}
BIT DI PARITA' P_1 - P_6
CIRCUITO DI CORREZIONE DEGLI ERRORI
CORRETTI D_0 - D_{15}
ABILITAZIONE D'USCITA
DBIN O R/$\overline{W}$
16
10
CODICE DELL'ERRORE
10 LED
BUS DATI
(CRI - CR10)

che sono i bit sottolineati come segue:

$$\underline{1}1\underline{1}1\quad \underline{0}0\underline{00}\quad \underline{1}1\underline{1}1\quad \underline{0}0\underline{1}1$$

Questa serie di bit contiene cinque 1: perciò verrà memorizzato uno 0 in P_2 in modo tale da mantenere il numero dispari di 1 che già esiste. Analogamente il bit P_3 verifica la parità dei bit D_{15}, D_{14}, D_{11}, D_{10}, D_7, D_6, D_3 e D_2 che sono sottolineati nell'esempio che segue:

$$\underline{11}11\quad \underline{00}00\quad \underline{11}11\quad \underline{00}11$$

Questa serie di bit contiene quattro 1: perciò verrà memorizzato un 1 in P_3 P_4 verifica la parità dei bit D_{15} - D_{12} e D_7 - D_4 i quali, nel codice che abbiamo utilizzato come esempio, contengono otto 1: perciò verrà memorizzato un 1 in P_4 P_5 verifica la parità dei bit D_{15} - D_8 e D_0 i quali, nell'esempio, contengono cinque 1, cosicché uno 0 verrà memorizzato in P_5. Infine P_6 verifica i bit D_{15}, D_{12}, D_{10}, D_9, D_7, D_4, D_2, D_1 e D_0 Questi bit sono sottolineati nel codice scelto come esempio che segue:

$$\underline{1}11\underline{1}\quad 0\underline{00}0\quad \underline{1}11\underline{1}\quad 0\underline{011}$$

Essi contengono sei 1: perciò P_6 memorizzerà un 1.

Generalmente, il generatore di parità è semplicemente un banco di 10 o 11 porte OR Esclusivo a due ingressi, per ogni bit di parità generato. Le linee dati passando attraverso i buffer a tre stati sono dirette alla riga di memoria e agli ingressi delle porte XOR corrispondenti del generatore di parità. Ciascuna uscita dei circuiti delle porte XOR viene inviata a quella parte della memoria riservata ai bit di parità (da P_1 a

La generazione di bit di parità viene svolta da una serie di porte XOR.

Figura 9.2. Tavola di generazione dei bit di controllo di parità per la memoria di figura 9.1.

Da D_{15} a D_0

Serie di bit dati in parallelo

BIT DATI																BIT DI CONTROLLO	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
							X	X	X	X	X	X*	X	X	X	P1	1
X		X		X		X	X	X		X		X*		X		P2	0
X	X			X	X			X	X			X*	X			P3	1
X	X	X	X					X	X	X	X					P4	1
X	X	X	X	X	X	X	X								X	P5	0
X			X		X	X		X			X		X	X	X	P6	1

* I bit P_1 P_2 e P_3 non corrispondenti indicano che il bit dati 3 è errato

** Ogni bit di controllo è di parità dispari per quei bit dati con X nella riga corrispondente

P_6) passando attraverso i buffer a tre stati. Perciò la generazione di parità viene compiuta automaticamente da circuiti relativamente semplici.

Individuazione e correzione degli errori

Un singolo errore nel bit dati si riscontra come tre o cinque errori dei bit di parità. Cercando i bit di parità errati può essere isolato il bit dati errato.

L'individuazione e la correzione degli errori si basano sul fatto che un qualsiasi bit dati sbagliato influirà su certi bit di parità, ma non necessariamente su tutti. In *figura 9.2* si noti che lo schema dei bit di parità che viene influenzato da un errore in un qualsiasi bit dati è unico per ciascuna posizione dei bit dati. Per esempio il bit dati D_3 influisce solo sui bit di parità P_1, P_2 e P_3. Perciò se D_3 fosse in errore, i bit di parità P_1, P_2 e P_3 sarebbero sbagliati, mentre gli altri bit di parità (P_4, P_5 e P_6) risulterebbero esatti effettuando una lettura di memoria e un controllo di parità. Nel codice dati utilizzato come esempio precedentemente, l'intero codice a 22 bit che si leggerebbe dalla memoria per un codice dati corretto sarebbe:

1111 0000 1111 0011 10 1101

Ora, se D_3 fosse in errore a causa di un circuito di memoria guasto nella riga o nel banco indirizzabile di circuiti di memoria, il codice ricevuto sul bus dati insieme con il codice di parità generato sarebbe:

1111 0000 1111 $\underline{1}$011 $\underline{01}$ $\underline{0}$101

Si noti che i bit sottolineati sono diversi da quelli che erano stati memorizzati. Confrontando lo schema di parità letto dalla memoria, 10 1101, con lo schema di parità generato dal codice dati in memoria (01 0101), il circuito di correzione degli errori (blocco posto nella parte inferiore di *figura 9.1*) sarebbe in grado di correggere il bit D_3 cambiando il suo livello prima di inviarlo ai dati del processore. Perciò, l'individuazione e la correzione automatiche degli errori rimandano i dati al processore nella loro forma corretta, cosicché l'elaborazione continui normalmente.

Se due o più circuiti di memoria in una riga si guastano (2 errori per codice dati), oppure se si guasta uno dei circuiti di parità, la correzione automatica degli errori non è possibile. Se ciò accadesse la memoria dovrebbe segnalare al processore che necessita di immediata riparazione perché il suo funzionamento sia attendibile. L'esistenza di un errore in più di un bit nell'intero dato a 22 bit più il codice di parità viene individuata controllando la parità di tutto il codice a 22 bit. Se questa parità risulta corretta, deve essersi verificato un numero pari di errori: non è possibile correggerli e si dovrà compiere una riparazione. Se invece la parità dei 22 bit risulta sbagliata significa che si è verificato un numero dispari di errori; di solito però il circuito è progettato sul presupposto piuttosto fondato che si sia verificato un solo errore (piuttosto che tre, cinque, ecc.) in quanto è estremamente improbabile che tre o più circuiti di una data riga si guastino contemporaneamente.

Per vedere se l'intero codice a 22 bit ha la parità corretta si deve confrontare il bit P_6 con la parità del codice a 22 bit. Se questi non corrispondono la parità è sbagliata. Nel codice di esempio in cui si suppone un errore nel bit D_3

1111 0000 1111 1011

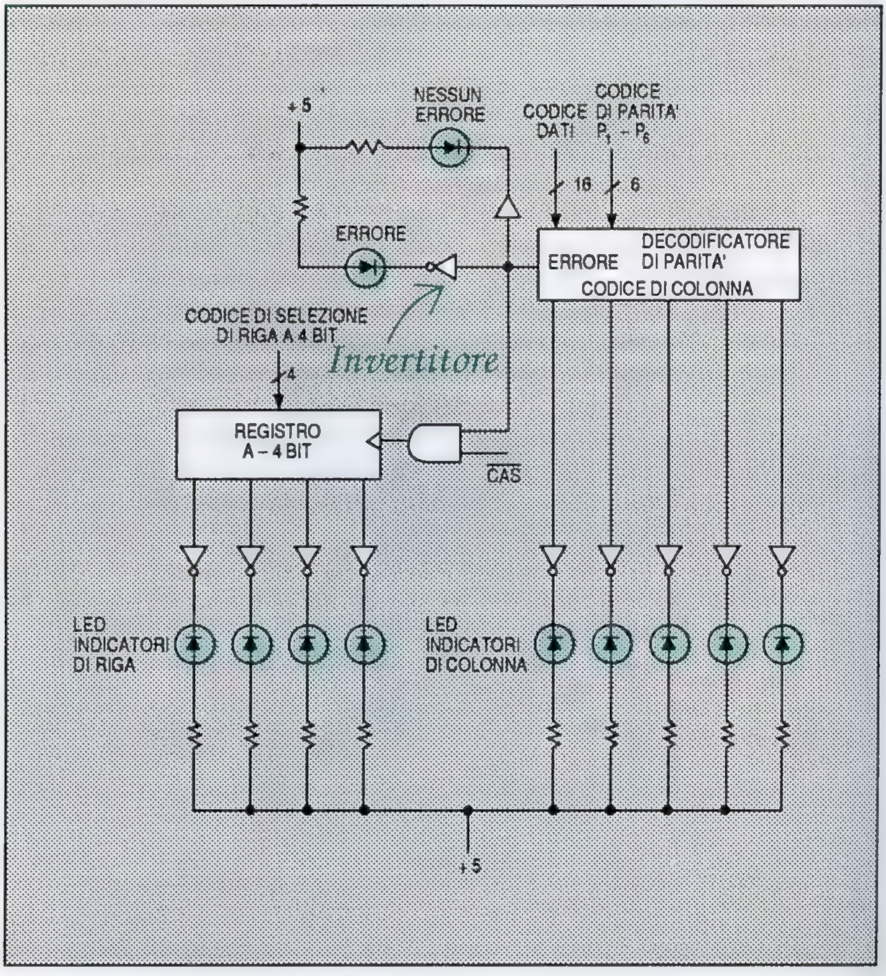
la parità del codice a 22 bit che si suppone sbagliato è 0 dato che in esso è contenuto un numero pari di 1. Poiché il bit P6 è un 1 (come determinato dal codice corretto), deve esserci un errore da qualche parte nel codice a 22 bit. Il circuito di correzione degli errori individuerà che questo si trova sulla linea D_3

Identificazione dell'errore

Il display di errore indica in codice binario quale riga di un particolare circuito integrato è guasta.

In *figura 9.3* è illustrato un semplice circuito di visualizzazione che utilizza i LED. Se non si è verificato alcun errore si accende il LED che segnala NO ERROR (Nessun errore). Quando il circuito che controlla la parità determina che si è verificato un errore, viene acceso il LED del display che segnala ERROR (Errore) e acquisisce il codice di riga nel registro a 4 bit. L'acquisizione si verifica sul fronte di salita del se-

Figura 9.3. Tipico circuito di visualizzazione degli errori


+5
NESSUN ERRORE
CODICE DATI
CODICE DI PARITA' $P_1 - P_6$
16
6
ERRORE
DECODIFICATORE DI PARITA'
ERRORE
CODICE DI COLONNA
CODICE DI SELEZIONE DI RIGA A 4 BIT
Invertitore
4
REGISTRO A - 4 BIT
CAS
LED INDICATORI DI RIGA
LED INDICATORI DI COLONNA
+5

gnale $\overline{\text{CAS}}$. L'uscita del registro è collegata ai quattro LED attraverso driver di corrente. La configurazione di questi LED indica in codice binario il numero della riga nella quale si è verificato l'errore. Se è avvenuto un errore l'uscita a 4 bit del decodificatore di parità nel circuito di visualizzazione dell'errore fornisce il codice binario che corrisponde al numero della colonna contenente il circuito guasto. La colonna potrebbe essere una delle colonne dati da D_0 a D_{15} o una delle colonne di bit di parità comprese tra P_1 e P_6. Osservando lo schema dei LED il riparatore potrà individuare il circuito integrato guasto.

Ricerca ed eliminazione dei guasti

Se è guasto più di un circuito di memoria un controllo della riga in questione tramite una semplice routine software dovrebbe rendere il sistema in grado di individuare con precisione quali circuiti della riga sono guasti. La routine sarà simile a quella trattata nel capitolo 6. Poiché in una riga sono contenute solo 64000 locazioni, questo test richiederà meno di un secondo al computer, supponendo che sia un computer il sistema che si sta provando. Se da questo test risultano guasti tutti i circuiti di una riga, allora molto probabilmente il problema è dovuto a un decodificatore o ad un filo scollegato nella linea di selezione della riga. Questo potrà essere individuato attraverso test a bassa frequenza del decodificatore e del cablaggio di selezione della riga che utilizzano la sonda e gli impulsatori logici o le misurazioni con il voltmetro, come descritto nel capitolo 6.

Guasto nel decodificatore

Se viene indicata guasta l'intera linea della memoria, allora: (1) può essere guasto il decodificatore, (2) può essere difettosa la linea di abilitazione del decodificatore, (3) la memoria può non essere ripristinata come dovrebbe e il guasto potrebbe trovarsi nel generatore del segnale di ripristino $\overline{\text{RAS}}$, (4) può essere difettosa l'alimentazione della memoria. Può essere controllata la presenza di ciascuno di questi guasti utilizzando sonde logiche o misurazioni con il voltmetro.

Guasto nell'unità di controllo del ripristino

Il modo più semplice per cercare un guasto nel circuito di controllo di ripristino consiste nell'utilizzare un oscilloscopio. I più probabili punti di guasto sono il contatore d'indirizzo, gli indirizzi di riga e l'oscillatore di ripristino.

La verifica del funzionamento del circuito di controllo del ripristino richiede la prova delle linee $\overline{\text{RAS}}$ del sistema in stato inattivo, cioè in un momento in cui il processore non stia compiendo alcuna operazione che richieda l'uso della memoria. L'impulso $\overline{\text{RAS}}$ dovrebbe verificarsi 256 volte ogni 2 millisecondi o, approssimativamente, una volta ogni 8 microsecondi. Questo test può essere compiuto mettendo una sonda di oscilloscopio sulla linea $\overline{\text{RAS}}$ ed esaminando anche le tre linee di indirizzo meno significative. Si dovrebbe osservare un tracciato rapidamente variabile sulle linee di indirizzo ed un treno di impulsi $\overline{\text{RAS}}$ ad una frequenza di 125 kilohertz approssimativamente. Analogamente, osservando le linee di indirizzo più significative, si dovrebbe individuare un segnale che cambia ogni millesimo di secondo. Naturalmente queste misurazioni mostreranno solo l'attività generale ad eccezione del treno di impulsi $\overline{\text{RAS}}$ uniformemente ripetitivo.

Se non si dovessero osservare questi segnali, l'unità di controllo del ripristino dovrebbe venire esaminata per verificare se il malfunzionamento avviene in essa. Può avvenire che il ciclo di ripristino non venga iniziato una volta ogni 8 microsecondi (se il multivibratore o l'oscillatore astabile ha smesso di funzionare) oppure può essere che il contatore di indirizzo non conti un modulo 256 (forse per un'apertura all'ingresso del clock del contatore o per un guasto nel contatore stesso). Potrebbe anche essersi verificata un'apertura sulla linea tra la linea $\overline{\text{RAS}}$ del circuito di ripristino e l'ingresso $\overline{\text{RAS}}$ del banco di memoria. Molti di questi guasti possono essere controllati tramite semplici controlli di continuità che vengono compiuti con voltmetri o con sonde logiche. L'uscita dell'oscillatore di ripristino può essere osservata con un oscilloscopio capace di misurare un'onda quadra di 125 kilohertz. Analogamente l'uscita del contatore può essere osservata con un oscilloscopio a più canali capace di misurare onde quadre nella gamma che va da 1 kilohertz a 125 kilohertz, a seconda di quale delle otto uscite del contatore si voglia osservare in un dato istante. I componenti fondamentali di un circuito di ripristino e la sua struttura sono già stati discussi nel capitolo 6.

Un analizzatore di firma permette l'analisi di lunghe sequenze di bit. Confronta la sequenza in questione con una sequenza standard nota.

TECNICHE DI ANALISI DI FIRMA

La semplice sonda logica descritta nei capitoli precedenti è limitata nella sua capacità di test a funzioni logiche sequenziali o combinatorie a bassa velocità e a test di continuità. Nei microcomputer i segnali su una data linea possono cambiare ad una

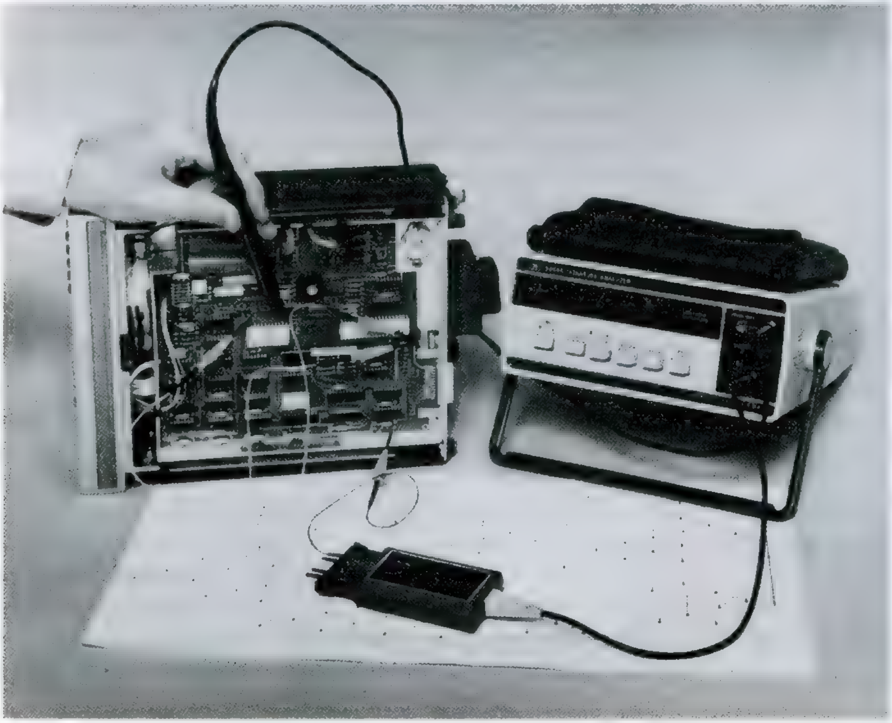


Figura 9.4. Analizzatore di firma *(Per gentile concessione della Hewlett Packard Co.)*

frequenza superiore a parecchi megahertz e non è quindi significativo registrare con una sonda un 1 o uno 0 in ogni dato istante.

È quindi necessario uno strumento che funga da sonda e sia più sofisticato, cioè uno strumento che sia in grado di determinare se una lunga sequenza di bit su di una linea sotto test è o meno corretta. Ciò significa che deve essere memorizzata una sequenza di bit perché sia confrontata con una sequenza standard sicuramente corretta oppure che una sequenza di bit deve essere compressa. In questo modo è possibile fornire un codice numerico di lunghezza determinata da confrontare con un codice che sia corretto. Questa è la strategia di base di uno strumento di prova noto come analizzatore di firma o multimetro di firma. Un modello di tale strumento è mostrato in *figura 9.4*. Si tratta di un'analizzatore di firma Hewlett Packard 5004A che può essere utilizzato per individuare e localizzare errori o guasti in un microcomputer o in una qualsiasi strumentazione digitale.

Individuazione degli errori

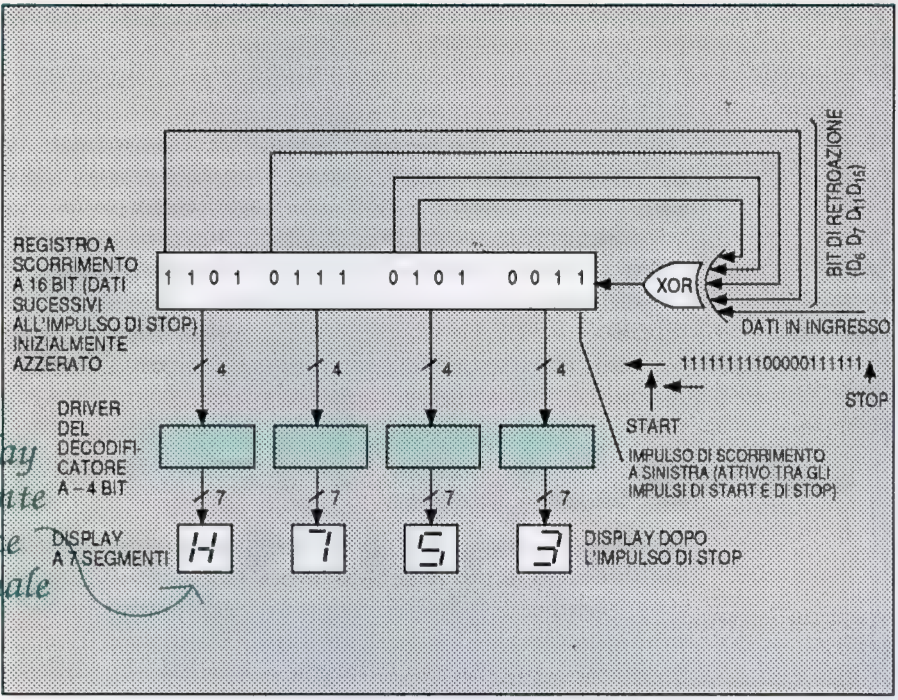
La principale operazione svolta dall'analizzatore di firma è quella di dividere una serie di bit binari per un polinomio noto. Il polinomio viene generato sulla base della composizione della serie di bit necessari per generare un codice binario a 16 bit che viene quindi visualizzato su quattro display a 7 segmenti. La divisione viene eseguita utilizzando un registro a scorrimento verso sinistra da 16 bit accoppiato con un addizionatore modulo 2.

È la stessa strategia di base utilizzata per individure gli errori nella comunicazione di serie di dati o nella lettura di floppy disk. Questa tecnica, essendo applicata nell'individuazione degli errori di dati nei floppy disk, è nota come controllo ciclico di ridondanza (CRC da Cyclic Redundancy Check). Quando un dato viene scritto nel disco, viene calcolato un codice CRC a 16 bit il quale viene a sua volta scritto alla fine di ciascun campo di memorizzazione del disco. Il numero CRC viene nuovamente calcolato quando il campo viene letto dal disco e il numero calcolato viene confrontato con quello memorizzato alla fine del campo. Se i due numeri CRC non coincidono, devono esserci uno o più errori nel dato che è stato letto.

L'analizzatore di firma utilizza la stessa tecnica di controllo di ridondanza ciclica per individuare gli errori nella memoria che viene utilizzata nella trasmissione delle comunicazioni di dati.

Lo stesso tipo di approccio viene utilizzato in *figura 9.5*, nella quale il registro a scorrimento emette il bit successivo proveniente dalla serie di dati nell'istante che viene specificato da un impulso di clock. Questo si verifica quando arriva un impulso di start e continua fino all'arrivo di un impulso di stop. Perciò viene trasferito nel registro un numero fissato di bit e il codice a 16 bit generato nel registro è unico per la sequenza di bit ricevuta. Se uno o più bit sono sbagliati, il numero memorizzato nel registro (e visualizzato sul display dello strumento) sarà diverso da quello osservato in una serie di bit che non contenga errori. Perciò, conoscendo il numero corretto a 16 bit e osservando quello del sistema sotto test, è possibile verificare il funzionamento di una linea o di un sistema. Occorre soltanto che siano noti i numeri per ogni linea di un sistema ben funzionante e che il sistema guasto venga fatto funzionare nello stesso identico modo di quello funzionante. Avendo l'esatto codice numerico di ogni linea del sistema, si potranno quindi sondare le linee di un sistema sotto test e determinare il loro stato operativo.

Figura 9.5. Funzionamento di base di un analizzatore di firma


BIT DI RETROAZIONE (D_6 D_7 D_{11} D_{15})
REGISTRO A SCORRIMENTO A 16 BIT (DATI SUCESSIVI ALL'IMPULSO DI STOP) INIZIALMENTE AZZERATO
1 1 0 1 0 1 1 1 0 1 0 1 0 0 1 1
XOR
DATI IN INGRESSO
4
11111111110000011111
START
STOP
DRIVER DEL DECODIFICATORE A – 4 BIT
IMPULSO DI SCORRIMENTO A SINISTRA (ATTIVO TRA GLI IMPULSI DI START E DI STOP)
7
DISPLAY A 7 SEGMENTI
H
7
5
3
DISPLAY DOPO L'IMPULSO DI STOP

Il codice del display è differente dal codice esadecimale

Figura 9.6. Codice del display di un analizzatore di firma *(Da: Gary Gordon and Horns Nordig, Electronics, 3 marzo 1977)*

GRUPPO DI 4 BIT DA VISUALIZZARE	CARATTERE VISUALIZZATO
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	C
1100	F
1101	H
1110	P
1111	U

Lettura del display

In *figura 9.5* il display mostra alcuni caratteri esadecimali (753) e una cifra che non è un carattere esadecimale standard. In questo caso si tratta di un H. Questo avviene perché i display a 7 segmenti non possono mostrare in maniera affidabile alcuni dei caratteri esadecimali (per esempio la B che potrebbe essere confusa con un 8). Infatti l'indicatore non fa necessariamente apparire sul display le cifre come i loro corrispondenti valori decimali. La tabella di *figura 9.6* mostra il codice esadecimale e il suo corrispondente carattere sul display. Poiché il display a 4 cifre (o caratteri) non rappresenta un codice di istruzioni specifico, ma è semplicemente uno schema di riferimento, l'assegnazione di un carattere a un codice esadecimale non è importante e può variare a seconda della casa costruttrice. Ecco perché sul display di *figura 9.5* appare H753 che corrisponde al codice 1101 0111 0101 0011 nel registro a scorrimento.

Ricerca dei guasti con un analizzatore di firma

Per utilizzare uno strumento del tipo mostrato in *figura 9.4* si hanno a disposizione due metodi di prova di uso comune: quello gestito dal software ("software-driven") e quello a ruota libera ("free-running").

Metodo gestito dal software

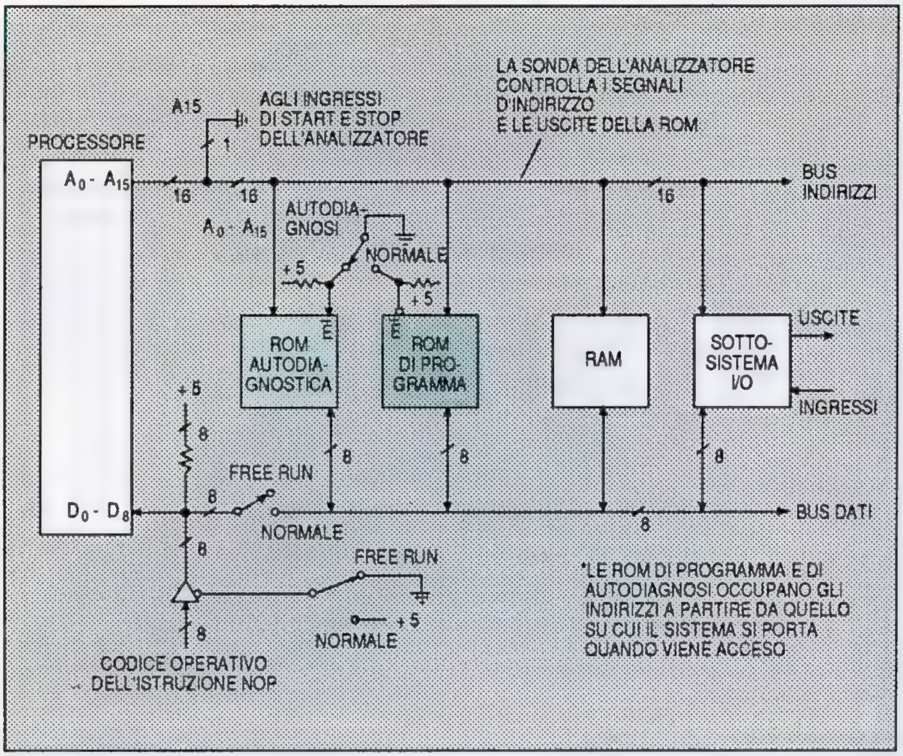
Il metodo gestito dal software utilizza un programma di test memorizzato in una parte della ROM del microcomputer che può essere eseguito per fornire sequenze note di bit sulle linee collegate alla RAM e ai dispositivi di I/O.

Metodo a ruota libera

Utilizzando un analizzatore di firma, le memorie ROM guaste di un sistema, le cui uscite sono isolate dal bus dati, emettono codici dati che vengono confrontati con codici noti provenienti da un'unità sicuramente funzionante. Una qualsiasi differenza viene analizzata per identificare la locazione specifica del guasto.

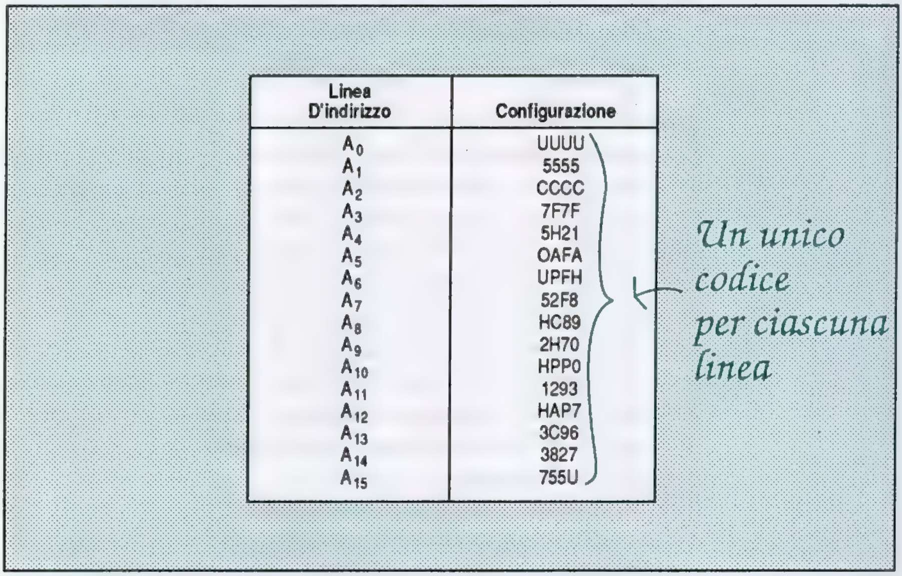
Nel metodo a ruota libera le linee dati del processore vengono scollegate dal bus dati del sistema. Ciò può essere fatto tramite dei cavallotti progettati già all'interno dell'hardware, tramite uno speciale zoccolo su cui inserire il microprocessore oppure utilizzando le tecniche di isolamento dei piedini. Le uscite della ROM vengono isolate dal bus dati del sistema in modo analogo e sul bus dati del processore si pone un singolo byte (8 bit) o una singola parola (16 bit). Solitamente l'istruzione è NOP (che significa No Operation) che non esegue alcuna operazione. Il processore si comporta come se avesse trovato un NOP ad ogni indirizzo in memoria, per cui incrementa l'indirizzo delle istruzioni e compie un altro fetch. Esso continuerà così (come mostrato in *figura 9.7*) a fornire tutti i possibili codici di indirizzo sul bus finché non viene tolta l'alimentazione del sistema.

Il clock del processore o l'impulso di lettura in memoria possono essere forniti all'analizzatore di firma mentre l'ingresso del clock e la transizione da 1 a 0 sulla linea corrispondente all'indirizzo più significativo possono essere utilizzati come segnali di start o di stop. Una sequenza nota o fissata di bit apparirà quindi su ciascuna delle linee di indirizzo e sulle uscite di ciascun piedino delle ROM (quando queste sono abilitate). In *figura 9.8* sono mostrati i codici che apparirebbero sull'analizzato-


LA SONDA DELL'ANALIZZATORE
CONTROLLA I SEGNALI
D'INDIRIZZO
E LE USCITE DELLA ROM
A15
AGLI INGRESSI
DI START E STOP
DELL'ANALIZZATORE
1
PROCESSORE
A0 - A15
16
16
16
A0 - A15
BUS
INDIRIZZI
AUTODIA-
GNOSI
NORMALE
+5
+5
E
E
ROM
AUTODIA-
GNOSTICA
ROM
DI PRO-
GRAMMA
RAM
SOTTO-
SISTEMA
I/O
USCITE
INGRESSI
+5
8
8
8
8
FREE RUN
D0 - D8
8
NORMALE
8
BUS DATI
8
FREE RUN
NORMALE
+5
8
CODICE OPERATIVO
DELL'ISTRUZIONE NOP

*LE ROM DI PROGRAMMA E DI AUTODIAGNOSI OCCUPANO GLI INDIRIZZI A PARTIRE DA QUELLO SU CUI IL SISTEMA SI PORTA QUANDO VIENE ACCESO

Figura 9.7. Configurazione del circuito per l'analisi di firma



Linea D'indirizzo	Configurazione
A_0	UUUU
A_1	5555
A_2	CCCC
A_3	7F7F
A_4	5H21
A_5	0AFA
A_6	UPFH
A_7	52F8
A_8	HC89
A_9	2H70
A_{10}	HPP0
A_{11}	1293
A_{12}	HAP7
A_{13}	3C96
A_{14}	3827
A_{15}	755U

Un unico codice per ciascuna linea

Figura 9.8. Configurazioni per il corretto funzionamento dei segnali d'indirizzo dei processori a 16 bit *(Da: Gary Gordon and Horns Nordig, Electronics, 3 marzo 1977)*

re di firma per un processore ad 8 bit. Confrontando i codici osservati su ciascuna linea di indirizzo con i codici di *figura 9.8*, è possibile verificare il funzionamento del processore e del bus indirizzi.

Ora, cambiando i segnali di start e di stop all'uscita del decodificatore (che gestisce la selezione del chip di una data ROM) e provando la sequenza all'uscita di ogni ROM, è possibile verificare ogni linea della ROM. Confrontando queste uscite col codice standard riportato nei manuali della strumentazione sotto test o ricavato da un'unità dello stesso tipo, è possibile verificare il funzionamento dell'intero sistema ROM. Analogamente si potranno verificare le uscite di tutti i decodificatori mentre il processore passa attraverso la sequenza di tutti i codici di indirizzo. Una volta completato il test a ruota libera saranno stati verificti anche gli stati in cui si trovano il processore, il bus indirizzi, i decodificatori di indirizzo e la ROM in modo tale che il riparatore saprà quale componente o quale insieme di linee è guasto e necessita di essere riparato.

Metodo autodiagnostico

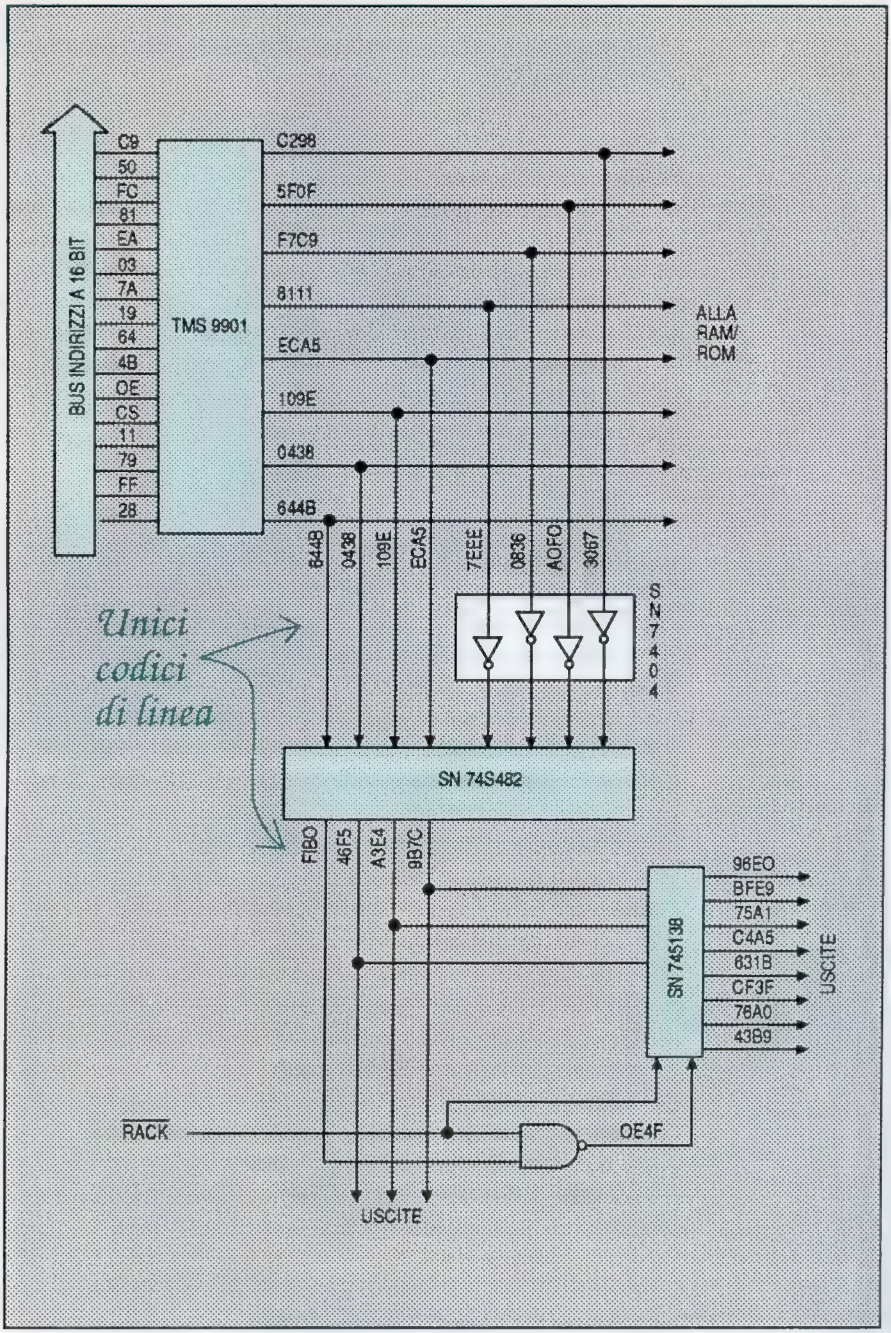
Il passo successivo che si può compiere nei sistemi che sono stati progettati in modo tale da poter essere esaminati con un analizzatore di firma è quello di attivare la ROM "autodiagnostica" e alimentare il processore dopo aver ricollegato il bus dati. Nel funzionamento a ruota libera le uscite della ROM erano isolate oppure le abilitazioni della RAM e di I/O erano inattive. Nel metodo autodiagnostico questi dispositivi devono essere ricollegati col sistema sotto test. Poiché è noto che il processore e le ROM funzionano correttamente, non si deve utilizzare il processore per eseguire sottoprogrammi che generino un impulso di start o di stop. Inoltre non dovrebbe essere utilizzato nemmeno per generare sequenze note sul bus dati o sul bus indirizzi tra gli istanti di start e di stop. Quindi scrivendo una serie di 1 e di 0 sotto forma di codici di test per la RAM e rileggendoli con l'analizzatore di firma collegato con una data RAM o ad una linea I/O, si può verificare il funzionamento di quella parte di sottosistema della RAM o di I/O. Anche in questo caso le cifre che si osservano sul display devono essere uguali a quelle riportate sul manuale del sistema oppure a quelle fornite da un sistema identico funzionante. Combinando il metodo a ruota libera e il metodo autodiagnostico con l'analizzatore di firma si potrà verificare tutto il sistema, compresi tutti i bus e tutti i dispositivi. Se si verifica un guasto, questo può essere circoscritto a un dato bus, ad un componente oppure ad una parte di un componente osservando quali firme sono sbagliate e quali sono corrette. Come esempio di documentazione che dovrebbe essere fornita nel manuale di un sistema, la *figura 9.9* mostra i codici esadecimali che si dovrebbero osservare su di un display per un ipotetico decodificatore di memoria di un sistema.

Nella modalità autodiagnostica, una ROM con una sequenza di autotest verifica il processore, la RAM e i circuiti di I/O, mentre un analizzatore di firma visualizza su di un display i codici dei segnali sui bus di collegamento. Una qualsiasi discrepanza dalle configurazioni note viene utilizzata per isolare l'errore.

Conclusione sull'analizzatore di firma

L'analizzatore di firma è uno strumento di test molto efficace in relazione al prezzo, che è in grado di identificare con facilità i guasti nei sistemi digitali complessi e nei microcomputer senza dover investire somme di denaro nelle più costose strumentazioni di ricerca dei guasti che verranno descritte più avanti in questo capitolo. L'a-

Figura 9.9. Modalità con cui un manuale d'istruzioni riporta tipicamente i codici di firma esatti


BUS INDIRIZZI A 16 BIT
C9
50
FC
81
EA
03
7A
19
64
4B
OE
CS
11
79
FF
28
TMS 9901
C298
5F0F
F7C9
8111
ECA5
109E
0438
644B
ALLA RAM/ ROM
644B
0438
109E
ECA5
7EEE
0836
AOFO
3067
SN 7404
Unici codici di linea
SN 74S482
FIBO
46F5
A3E4
9B7C
SN 745138
96EO
BFE9
75A1
C4A5
631B
CF3F
76A0
43B9
USCITE
RACK
OE4F
USCITE

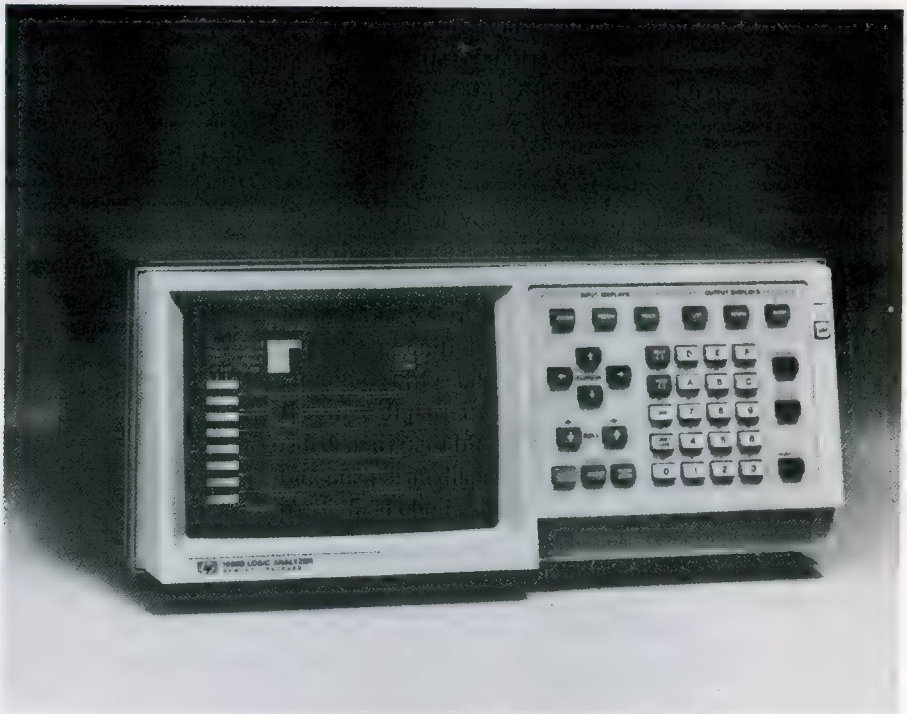
nalizzatore di firma è fondamentalmente un rivelatore di errori, ad una sola sonda, per sistemi che producono una lunga sequenza di serie note di bit. Le strumentazioni di test più costose permettono di effettuare la ricerca dei guasti di un sistema in parecchi modi diversi. Tra questi ultimi sono inclusi il mostrare sugli schermi CRT le forme d'onda del sistema, la memorizzazione e la stampa di codici esadecimali in sequenza e il controllo automatico di sistemi per mezzo di un computer. L'analizzatore logico è il meno costoso di questi strumenti di test più complessi e verrà trattato qui di seguito.

L'analizzatore logico è un tipo di oscilloscopio che può memorizzare segnali temporali osservati, controllare più segnali contemporaneamente, e osservare le sequenze di istruzioni che risultano da una specifica combinazione di segnali.

ANALIZZATORI LOGICI

Mentre l'oscilloscopio a più canali può essere utilizzato per determinare l'andamento generale dei segnali, come, per esempio, l'esistenza di un ciclo di ripristino periodico, con uno strumento simile non è possibile congelare completamente le informazioni dalla memoria o captare degli errori momentanei nei segnali in una memoria del tipo mostrato in *figura 9.1*. Inoltre, nei microcomputer è spesso necessario visualizzare molti segnali contemporaneamente e, possibilmente, dopo che si siano verificate nel sistema alcune particolari combinazioni di segnali. Ancora: si può deside-

Figura 9.10. Tipico analizzatore logico che mostra le forme d'onda di temporizzazione *(Per gentile concessione della Hewlett Packard Co.)*



rare di determinare la sequenza di istruzioni che si verifica dopo che sono state osservate alcune combinazioni di segnali. Un oscilloscopio a più canali non può soddisfare i requisiti di tali misurazioni, mentre ciò è possibile per uno speciale strumento noto come analizzatore logico. Il funzionamento generale di un analizzatore logico è relativamente semplice, anche se, di solito, si tratta di uno strumento abbastanza complesso. In *figura 9.10* è mostrato un tipico analizzatore logico che presenta sullo schermo forme d'onda di temporizzazione. Il diagramma a blocchi di un semplice analizzatore logico è mostrato invece in *figura 9.11*.

Campionamento

La sezione d'ingresso è composta da un clock di campionamento che guida un circuito di campionamento e mantenimento (sample and hold). Il circuito determina quando il codice digitale in ingresso proveniente da ciascuna sonda di prova viene campionato e memorizzato dall'analizzatore. Un analizzatore potrebbe avere, per esempio, un clock di campionamento che funzioni con la frequenza di 100 megahertz (100 milioni di campionamenti al secondo) il che significa un campionamento ogni 10 nanosecondi (ns). Poiché la forma d'onda che appare sullo schermo proviene dal circuito sample and hold, la transizione avviene con ritardo invece che in tempo reale come avveniva in un oscilloscopio. L'entità di questo ritardo dipende dalla frequenza del campionamento: più elevata sarà la frequenza, minore sarà il ritardo e, di conseguenza, l'errore. L'errore viene dato come scarto in più o in meno rispetto al tempo reale della transizione. Perciò la precisione dello strumento che utilizza una frequenza di 100 megahertz avrà una gamma di errore possibile da -10 ns a +10 ns. Alcuni analizzatori hanno un clock di campionamento con la frequenza di 400 megahertz che fornisce una maggiore precisione avendo un errore nella gamma da -2,5 ns a +2,5 ns. Ciò significa che una transizione di fronte da 0 ad 1 o da 1 a 0 può essere determinata entro un errore di 5 ns se l'analizzatore ha un clock con la frequenza di 400 megahertz. Ciò è più che sufficiente per la maggior parte dei sistemi moderni basati sui microprocessori.

Canali di ingresso

A seconda del costo dello strumento, l'analizzatore può fornire un numero di canali d'ingresso o sonde che va da 16, come indicato in *figura 9.11*, a più di 40. Ciascun canale è collegato con una linea separata: perciò un dispositivo d'ingresso a 16 canali può analizzare tutte le linee di un bus dati a 16 linee contemporaneamente.

Memoria

Per svolgere le sue molte funzioni, un analizzatore logico deve avere una memoria molto grande.

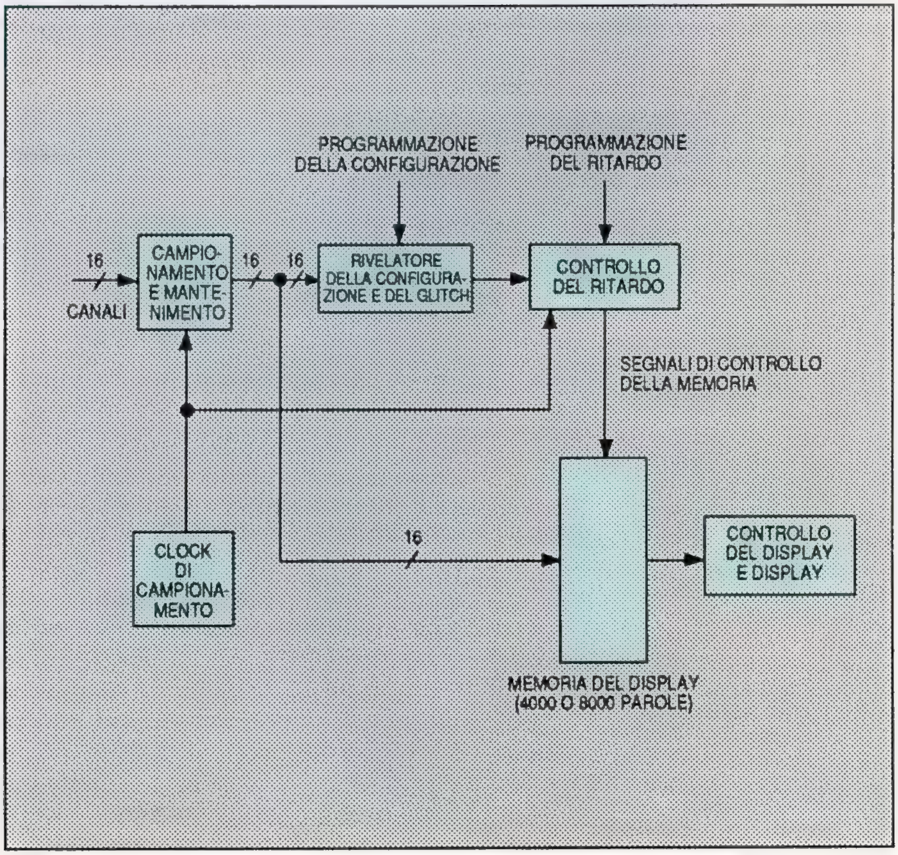
Allo scopo di mostrare le forme d'onda dei canali su di uno schermo CRT devono essere memorizzati nell'analizzatore i campionamenti più recenti. Il prodotto della lunghezza della memoria per il periodo del clock di campionamento fornisce il tempo totale rappresentato dalla larghezza dello schermo di visualizzazione. Se, per esempio, gli 8000 campionamenti più recenti sono memorizzati nell'analizzatore ed esso sta funzionando con un clock di campionamento che ha un periodo di 10 nanosecondi, l'ampiezza dello schermo rappresenta un tempo totale di 80.000 nanosecondi o 80 microsecondi. Per visualizzare un intervallo di tempo più lungo si dovrebbe diminuire la frequenza del clock di campionamento. Quindi, utilizzando ancora la stessa memoria di campionamento da 8.000 campioni (per ogni canale che si sta analizzando), ma con un clock che ha una frequenza di 10 megahertz, l'ampiezza totale dello schermo rappresenterebbe 8.000 volte 100 nanosecondi (0,1 microsecondi) o 800 microsecondi. Questo potrebbe rappresentare il tempo necessario per compiere 100 ripristini di memoria o 400 cicli di istruzioni del microprocessore. Se l'analizzatore è in grado di analizzare 16 canali e di conservare 4000 campioni per canale, la memoria totale dell'analizzatore sarà di 64.000 bit d'informazione o di 4000 parole. Questa non è una memoria molto grande, ma può essere molto costosa dato che deve essere costruita con dispositivi ad alta velocità in grado di memorizzare i campioni alla frequenza di 100 megahertz.

Quando si verifica il funzionamento di una memoria utilizzando un analizzatore logico, l'operatore può controllare a che punto o sotto che condizioni la memoria immagazzina i successivi campioni di segnale.

Display

Dato che la memoria dell'analizzatore deve, necessariamente, essere limitata, come è possibile determinare quali sono i 4000 campioni che stanno apparendo sullo schermo? Generalmente l'analizzatore è programmabile per raccogliere i campioni imme-

Figura 9.11. Schema a blocchi di un analizzatore logico a 16 canali


PROGRAMMAZIONE DELLA CONFIGURAZIONE
PROGRAMMAZIONE DEL RITARDO
16
CANALI
CAMPIONAMENTO E MANTENIMENTO
16
16
RIVELATORE DELLA CONFIGURAZIONE E DEL GLITCH
CONTROLLO DEL RITARDO
SEGNALI DI CONTROLLO DELLA MEMORIA
16
CLOCK DI CAMPIONAMENTO
CONTROLLO DEL DISPLAY E DISPLAY
MEMORIA DEL DISPLAY (4000 O 8000 PAROLE)

diatamente, dopo un ritardo specificato o dopo che si sono verificate alcune combinazioni di segnali. Ciò fa sì che l'utente sia in grado di specificare con estrema precisione che intervalli di tempo di funzionamento del sistema desidera in funzione di qualche caratteristica dei segnali che si stanno analizzando. Ne risulta che l'analizzatore deve avere dei rivelatori di combinazioni di segnali e dei contatori di ritardo che possono essere programmati attraverso i controlli sul pannello frontale. Questi circuiti di rivelazione e di conteggio forniscono i segnali che portano la memoria ad immagazzinare i campioni successivi. I circuiti del display forniscono la visualizzazione di base disegnando la forma d'onda che mostra i livelli 0 ed 1 sul CRT per ogni canale che si sta sondando.

Rivelatore di glitch

Lo strumento può anche possedere dei rivelatori di impulsi spuri (glitches) che individueranno con precisione gli errori di segnali quali gli 0 o gli 1 momentanei che si suppone non dover essere presenti. (Un glitch è una transizione di segnale non desiderata solitamente dovuta al rumore.) Generalmente si scopre che si tratta di transizioni da 0 a 1 a 0 oppure da 1 a 0 a 1 che avvengono in un intervallo di tempo particolarmente breve e specificato (per esempio l'intervallo fra due o più campionamenti successivi). Se, per esempio, i normali cambiamenti di segnale non dovessero verificarsi che una volta ogni 100 nanosecondi, lo strumento potrebbe essere programmato per indicare come glitch ogni transizione da 0 a 1 a 0 oppure da 1 a 0 a 1 che accade in meno di 40 nanosecondi.

La capacità di programmare istruzioni e di memorizzare e visualizzare sul display un gran numero di forme d'onda del sistema conferisce all'analizzatore logico molte caratteristiche potenziali che l'oscilloscopio a più canali non ha. Inoltre il fatto che l'analizzatore logico sia in grado di individuare e visualizzare sul display anche eventi di segnali asincroni lo rende più versatile dell'oscilloscopio, il quale è, invece, il migliore nel visualizzare sul display eventi ripetitivi o sincroni.

ESEMPIO DI ANALIZZATORE LOGICO

La *figura 9.12* mostra una forma d'onda come la si potrebbe osservare sullo schermo di un analizzatore logico. Per semplificare le cose sono mostrati solo pochi canali. Perché si abbia un tale display si suppone che l'analizzatore sia collegato per misurare i segnali di ripristino forniti da un'unità di controllo di ripristino del tipo discusso precedentemente. Il processore del sistema è stato rimosso o disabilitato elettricamente dalla memoria applicando una tensione tale da portare la sua linea HOLD nel suo stato attivo. La linea HOLD viene attivata da un multivibratore monostabile e da una combinazione di interruttori nell'analizzatore logico per mantenere il processore in una condizione di inattività durante tutto il tempo attivo del display. Questo fa sì che il processore porti in uno stato di alta resistenza o di disabilitazione i buffer a tre stati su tutti i suoi segnali di controllo, di indirizzi, di dati e di memoria e impedisce al processore di tentare l'accesso in memoria. Perciò tutta l'attività della memoria sarà controllata soltanto dall'unità di controllo di ripristino.

Programmazione dello strumento

Il fatto che l'analizzatore logico sia programmabile permette all'operatore di selezionare i segnali che devono essere analizzati e i livelli iniziali dai quali essi devono partire.

La frequenza di campionamento è fissata a 100 megahertz in modo tale da fornire un periodo di campionamento di 10 nanosecondi e un tempo totale di display di circa 40 microsecondi per una memoria da 4000 parole. Questo permette all'analizzatore di visualizzare sul display il verificarsi di circa 5 cicli di ripristino. I segnali scelti per il display sono $\overline{RAS}$, $\overline{CAS}$ (che dovrebbe rimanere alto per l'attività di ripristino), RFSH CLK (il clock che attiva il ciclo di ripristino compiendo una transizione dal livello basso al livello alto una volta ogni 8 microsecondi) e le tre linee di indirizzo meno significative per l'indirizzo di riga.

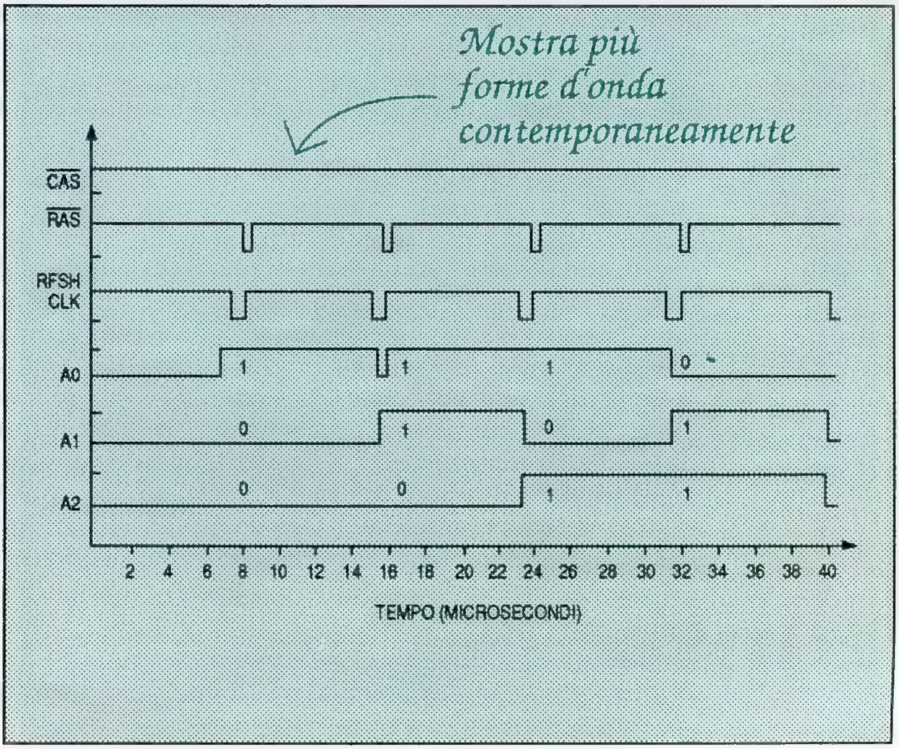
La memorizzazione e la visualizzazione sul display delle informazioni catturate dall'analizzatore sono programmate in modo tale da iniziare quando si verificano contemporaneamente le seguenti condizioni:

1. Il segnale HOLD è attivo.
2. Le tre linee di indirizzo meno significative sono a 0.
3. Il clock compie una transizione dal livello basso al livello alto.

Questa capacità di programmazione permette all'utente di comunicare all'analizzatore quali segnali utilizzare e quali livelli dovranno avere questi ultimi quando deve iniziare la visualizzazione sul display. Perciò l'analizzatore fa apparire sul display solo i blocchi di segnale che interessano.

Poiché i segnali del sistema vengono campionati solo secondo i tempi del clock di campionamento dell'analizzatore, il display indica i cambiamenti nei livelli di un segnale soltanto in corrispondenza dei tempi di campionamento, anche se il reale se-

Figura 9.12. Display di un analizzatore logico - Esempio di un controllore di ripristino



Mostra più forme d'onda contemporaneamente gnale del sistema può essere cambiato in un qualsiasi momento tra il precedente impulso del clock di campionamento e quello che mostra l'avvenuto cambiamento. (Si faccia riferimento alla trattazione riguardante l'errore di campionamento.) L'analizzatore determina se il segnale è uno 0 o un 1 nell'istante di campionamento confrontando questo livello di tensione con un certo livello di tensione programmato all'interno dell'analizzatore dall'operatore per vedere se è superiore o inferiore a quest'ultimo. Nei circuiti TTL l'operatore normalmente dovrebbe programmare una soglia di tensione di 1,4 volt. Altre famiglie logiche possono invece richiedere una soglia differente: per esempio i circuiti ECL potrebbero richiedere una soglia di 0 volt. (Si faccia riferimento al capitolo 1 per quanto riguarda la trattazione delle famiglie logiche.)

Significato del display

Una volta capita la relazione che intercorre tra le varie forme d'onda e i loro tracciati sullo schermo, è possibile determinare che cosa stia rivelando la forma d'onda riguardo al funzionamento del sistema.

Controllo dei segnali $\overline{RAS}$ e $\overline{CAS}$

I primi segnali da controllare sono le forme d'onda $\overline{RAS}$ e $\overline{CAS}$. Il segnale $\overline{CAS}$ dovrebbe rimanere al livello 1 durante tutta la visualizzazione sul display come avviene in *figura 9.11*. Finché esso resta alto, non c'è alcun guasto che sta portando erroneamente $\overline{CAS}$ ad un livello basso. Ciò è importante perché se $\overline{CAS}$ si abbassasse in questo intervallo di tempo farebbe sì che uno o più circuiti su una certa linea dati applichino dei livelli di tensione a quella stessa linea. Questa tensione potrebbe provocare un guasto in uno o più circuiti collegati con quella linea dati.

Capendo le relazioni che intercorrono tra i segnali nella memoria dinamica, diventano evidenti le relazioni di causa ed effetto.

La linea $\overline{RAS}$ dovrebbe avere degli impulsi di livello basso una volta ogni 8 microsecondi per circa 250 nanosecondi, come avviene in *figura 9.12*. Se la linea $\overline{RAS}$ è bloccata a 0 o ad 1, o se gli impulsi non venissero generati ad una frequenza di 125 kilohertz (ogni 8 microsecondi), l'intera memoria non funzionerebbe più correttamente. In questo caso il guasto dovrebbe essere sulla linea che va dall'uscita $\overline{RAS}$ del controllore al banco di memoria, oppure dovrebbe essere guasto il circuito che genera il segnale $\overline{RAS}$. Se viene utilizzata un'unità di controllo di ripristino come quella in *figura 6.9* ma con la temporizzazione mostrata in *figura 6.8*, il segnale $\overline{RAS}$ viene generato da un multivibratore monostabile che emette un impulso basso per 250 nanosecondi. Il multivibratore viene attivato dal fronte di salita (transizione dal livello basso al livello alto) del clock di temporizzazione del ripristino a 125 kilohertz. Se il segnale $\overline{RAS}$ proveniente dall'unità di controllo è bloccato ad 1, il multivibratore è guasto oppure c'è un'apertura o un corto circuito sulla linea di ingresso del clock del multivibratore oppure il clock di temporizzazione del ripristino è non operativo.

Controllo dell'indirizzamento e della temporizzazione

Poiché in *figura 9.12* i segnali $\overline{RAS}$ e $\overline{CAS}$ funzionano normalmente, i segnali che devono essere controllati successivamente sono le uscite A_0, A_1 e A_2 del contatore e

Osservando i segnali d'indirizzo e i segnali di tempificazione è possibile verificare se i maggiori sottosistemi stanno funzionando correttamente.

il clock di temporizzazione del ripristino. Per prima cosa il clock dovrebbe essere un treno di impulsi a 125 kilohertz, perché possa produrre la temporizzazione mostrata in *figura 6.8*. Il segnale dovrebbe essere alto per 7,5 microsecondi e basso per 0,5 microsecondi. Durante il ritorno ad un livello alto, il segnale $\overline{RAS}$ dovrebbe divenire basso per un periodo di 0,25 microsecondi. Il buffer a tre stati all'uscita dovrebbe essere abilitato per permettere al contatore di generare gli indirizzi delle righe di ripristino da A0 ad A7 in modo da collegare i suoi segnali con le linee di indirizzo di riga della memoria di *figura 9.1*.

L'osservazione dei segnali di indirizzo A_0, A_1 e A_2 e delle forme d'onda del clock di temporizzazione in *figura 9.12* sembra indicare che questa parte dell'unità di controllo del ripristino sta funzionando normalmente. Il clock di temporizzazione sta generando un impulso attivo basso ripetitivo di circa 0,5 microsecondi una volta ogni 8 microsecondi e i segnali di indirizzo provenienti dall'unità di controllo sono attivi durante questo periodo e durante l'intervallo di 0,25 microsecondi in cui il segnale $\overline{RAS}$ si trova al suo livello basso. La transizione dal livello basso al livello alto del clock sembra coincidere perfettamente con la generazione dell'impulso basso attivo $\overline{RAS}$: perciò sembra che il generatore di clock e il controllo del buffer a tre stati funzionino correttamente.

Il cambiamento delle sequenze d'indirizzi attraverso i codici, incrementati di 1, indica il funzionamento corretto o non corretto di un generatore d'indirizzo di ripristino.

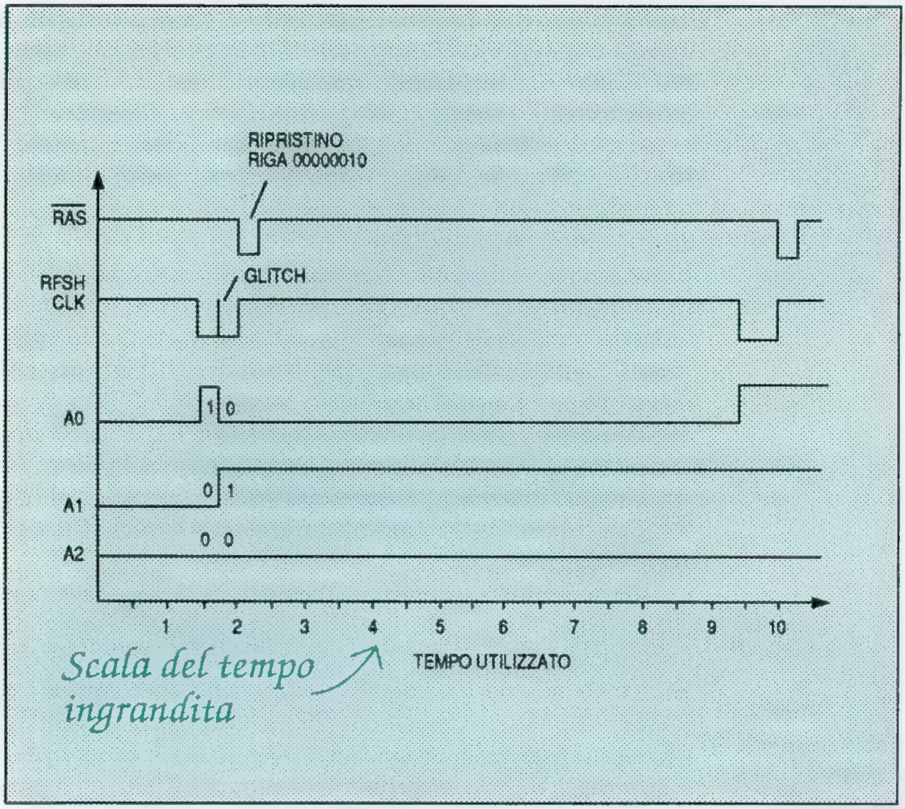
L'unità di controllo del ripristino sembrerebbe dunque funzionare correttamente, il che dovrebbe indicare che la ragione del guasto della memoria debba trovarsi in un'altra parte del sistema. Comunque, prima di stabilire che l'unità di controllo funziona correttamente è necessario esaminare più attentamente i segnali di indirizzo. Il codice di indirizzo (in *figura 9.12* sono visualizzati sul display solo i tre bit meno significativi) dovrebbe aumentare di 1 nel codice binario, a partire da 000, per ciascun impulso $\overline{RAS}$ appena prima il fronte attivo di $\overline{RAS}$ (ci si riferisca alla temporizzazione riportata in *figura 6.8*). Perciò il codice di indirizzo ($A_2A_1A_0$) dovrebbe diventare 001 appena prima del primo segnale $\overline{RAS}$, come mostrato in *figura 9.12*. Analogamente dovrà diventare 010 appena prima del secondo segnale $\overline{RAS}$. Comunque, esaminando il codice di indirizzo appena prima del secondo segnale $\overline{RAS}$, si vede che esso diventa dapprima 010 e quindi 011 durante il periodo in cui è basso il clock di temporizzazione del ripristino, prima che si verifichi il secondo segnale $\overline{RAS}$. Perciò la riga 010 viene saltata. Quindi il codice di indirizzo passa da 010 a 101 a causa del terzo impulso $\overline{RAS}$, saltando la riga 100. Durante il quarto impulso $\overline{RAS}$ esso diviene 110 come dovrebbe, ma poi, a causa del quinto impulso $\overline{RAS}$ diviene 000 saltando la riga 111.

Dunque un certo tipo di errore nel circuito sta provocando un avanzamento non corretto del conteggio degli indirizzi di riga durante gli intervalli di ripristino. Ciò fa sì che alcuni indirizzi di riga vengano saltati e che, senza ripristino, queste righe perdano i loro dati.

Controllo dei glitch

Questa situazione spiegherebbe il funzionamento errato di tutta la memoria anche se il circuito di ripristino dovesse sembrare correttamente funzionante. Per determinare la causa del problema (si dovrebbe sospettare un glitch o un rumore nel sistema), si può espandere il display dell'analizzatore in modo da mostrare due impulsi $\overline{RAS}$

Figura 9.13. Display ingrandito di un analizzatore - Controllore di ripristino



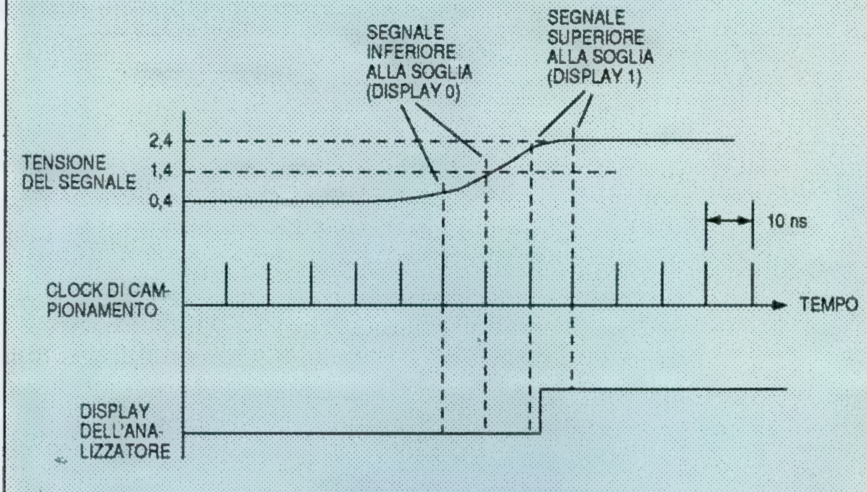
Quando si sospetta un errore di temporizzazione è possibile espandere nel tempo il display dell'analizzatore logico per osservare con più attenzione i segnali del sistema.

con il rivelatore di glitch abilitato. I glitch appaiono sul display come barre verticali. Ciascun glitch contiene una transizione dal livello alto al livello basso che, se abbastanza ampia, provoca un avanzamento del contatore e, quindi, il salto di uno o più indirizzi di riga.

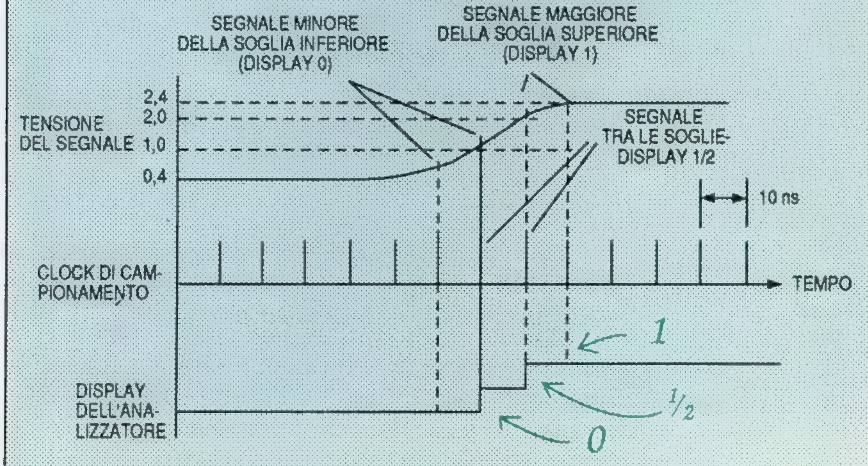
La *figura 9.13* mostra un display espanso in questo modo, con la presenza di un glitch appena prima il primo impulso $\overline{\text{RAS}}$, la quale provoca un incremento supplementare del codice di indirizzo che passa così da 010 a 011. Il glitch potrebbe essere provocato da un rumore, da un contatto guasto nel cablaggio dal generatore di clock all'ingresso del clock del contatore, o dallo stesso generatore di clock guasto.

L'uso di un analizzatore logico è l'unico modo pratico per individuare un problema di questo tipo in un sistema complesso. Anche se il display a scala espansa non avesse mostrato un glitch, l'errore sarebbe stato eventualmente rivelato da ripetute visualizzazioni sul display. Inoltre, se si dovesse sospettare la presenza di glitch, l'analizzatore potrebbe essere programmato in modo da iniziare il campionamento e la visualizzazione sul display quando viene individuato il primo glitch dal rivelatore di glitch.

Figura 9.14. Soglie alternative del display di un analizzatore


SEGNALE INFERIORE ALLA SOGLIA (DISPLAY 0)
SEGNALE SUPERIORE ALLA SOGLIA (DISPLAY 1)
TENSIONE DEL SEGNALE
2,4
1,4
0,4
10 ns
CLOCK DI CAMPIONAMENTO
TEMPO
DISPLAY DELL'ANALIZZATORE

a. Controllo e Display a Soglia Singola


SEGNALE MINORE DELLA SOGLIA INFERIORE (DISPLAY 0)
SEGNALE MAGGIORE DELLA SOGLIA SUPERIORE (DISPLAY 1)
SEGNALE TRA LE SOGLIE DISPLAY 1/2
TENSIONE DEL SEGNALE
2,4
2,0
1,0
0,4
10 ns
CLOCK DI CAMPIONAMENTO
TEMPO
DISPLAY DELL'ANALIZZATORE
1
1/2
0

b. Controllo e Display a Doppia Soglia

Ulteriori utilizzazioni dell'analizzatore logico

L'analizzatore logico è in grado di fornire ulteriori prestazioni che rendono più semplice il compito del riparatore. Esso infatti offre di solito almeno altre due possibilità di visualizzazione sul display.

Display a soglia doppia

Il display della soglia di livello permette di misurare il tempo richiesto da un segnale per cambiare tra due livelli di soglia.

Una di queste consiste nel visualizzare forme d'onda sul display utilizzando circuiti di ingresso a soglia singola o doppia. Questi sono molto utili per osservare relazioni temporali: per esempio il tempo che un segnale impiega per passare da 0 ad 1 o viceversa. In *figura 9.14a* è mostrato un segnale che varia lentamente e come questo viene visualizzato col metodo della soglia singola. Il lento tempo di salita non è evidente con una soglia singola di 1,4 volt. Utilizzando invece, per lo stesso segnale, una soglia doppia di 1,0 volt e 2,0 volt insieme al display di un analizzatore a tre livelli, si ottiene la forma d'onda mostrata in *figura 9.14b*. Il display a gradini indica il tempo impiegato dal segnale che varia per compiere una transizione da 0 ad 1.

Display con codici esadecimali e binari

I display dei codici esadecimale e binario permettono al riparatore di notare un cambiamento inaspettato nella progressione degli indirizzi e nella frequenza con cui si ripetono speciali segnali di ripristino.

Un'altra possibilità del display, che è molto utile per individuare velocemente alcuni problemi, è quella di far apparire sul display codici esadecimali e binari invece di forme d'onda per ogni specificata condizione di campionamento. Se, per esempio, in *figura 9.12* l'operatore avesse programmato l'analizzatore in modo tale da visualizzare sul display i due segnali a 4 bit che rappresentano il codice di indirizzo di riga ad 8 bit proveniente dal contatore su ogni transizione dal livello alto al livello basso del segnale $\overline{\text{RAS}}$, il display dovrebbe essere simile a quello di *figura 9.15*. Poiché il riparatore sa che ciascun codice esadecimale visualizzato sul display dovrebbe aumentare di 1 dall'alto verso il basso, un problema di indirizzamento dovrebbe essere facilmente individuato notando il salto di 2 del codice esadecimale durante il secondo, il terzo e il quinto impulso $\overline{\text{RAS}}$. Il riparatore potrebbe quindi optare per il modo di visualizzazione mostrato in *figura 9.13* per determinare la causa del problema. Il display mostrato in *figura 9.15* permette anche all'operatore di controllare velocemente che gli impulsi $\overline{\text{RAS}}$ si ripetano, come richiesto, uno ogni 8 microsecondi.

Analisi del programma

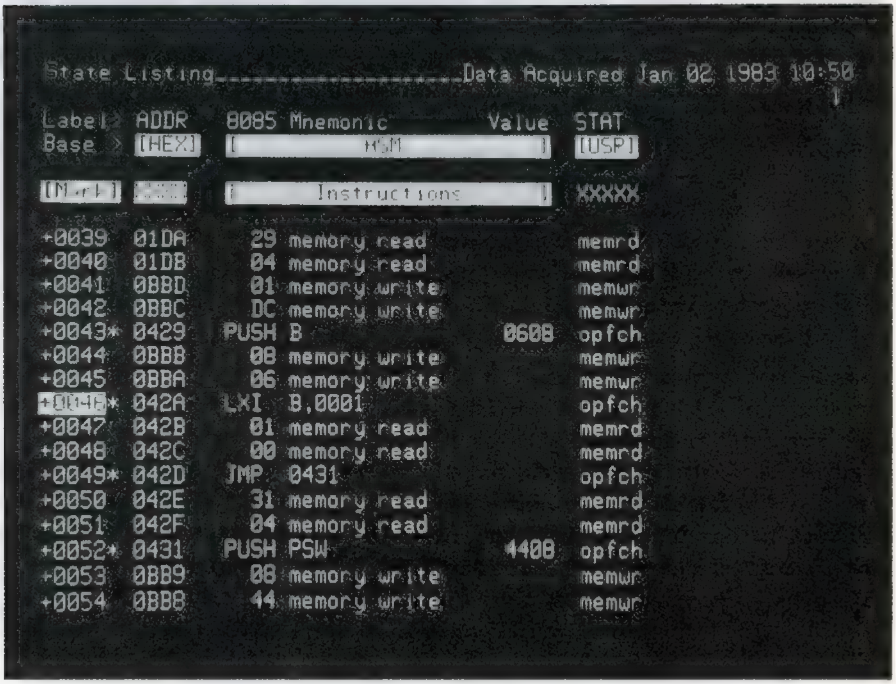
Alcuni analizzatori logici possiedono una strumentazione software o hardware che permette loro di ricondurre i codici binari al loro set di istruzioni (disassemblatore). Questa caratteristica può essere molto utile nel determinare le ragioni per cui nuovi progetti di sistemi non funzionano correttamente, così come le ragioni per cui si è verificato il mancato funzionamento di sistemi di ricerca dei guasti. In *figura 9.16* è presentato un esempio di un analizzatore logico che mostra i codici in linguaggio assemblativo per un microprocessore.

Con questo tipo di analizzatore le sonde di ingresso sono collegate con tutte le linee del bus indirizzi, con tutte le linee del bus dati e con le linee di temporizzazione e di controllo del microprocessore. Questo permette all'analizzatore di seguire, memo-

Figura 9.15. Display opzionale a codice esadecimale per un analizzatore logico

$\overline{\text{RAS}}$ Numero dell'impulso	Tempo in Microsecondi	Linee di indirizzo da A_0 a A_7
1	8.000	01
2	16.000	03
3	24.000	05
4	32.000	06
5	40.000	08
•	•	•
•	•	•

Figura 9.16. Analizzatore logico che mostra un codice in linguaggio Assemblativo *(Per gentile concessione della Hewlett Packard Co.)*



```
State Listing_______________________Data Acquired Jan 02 1983 10:50

Label > ADDR    8085 Mnemonic               Value  STAT
Base  > [HEX]   [            ASM           ]       [USP]

[Mark]          [        Instructions      ]       XXXXX

+0039   01DA      29 memory read                   memrd
+0040   01DB      04 memory read                   memrd
+0041   0BBD      01 memory write                  memwr
+0042   0BBC      DC memory write                  memwr
+0043*  0429    PUSH B                      0608   opfch
+0044   0BBB      08 memory write                  memwr
+0045   0BBA      06 memory write                  memwr
+0046*  042A    LXI  B,0001                        opfch
+0047   042B      01 memory read                   memrd
+0048   042C      00 memory read                   memrd
+0049*  042D    JMP  0431                          opfch
+0050   042E      31 memory read                   memrd
+0051   042F      04 memory read                   memrd
+0052*  0431    PUSH PSW                    4408   opfch
+0053   0BB9      08 memory write                  memwr
+0054   0BB8      44 memory write                  memwr
```

rizzare e visualizzare sul display le istruzioni che il microprocessore sta svolgendo nel suo linguaggio originario assemblativo. Ciò è molto utile quando si sta cercando di determinare se il programma nella memoria di programma (come viene eseguito dal processore) è, o meno, l'esatta sequenza di istruzioni. L'analizzatore può essere programmato in modo tale che cominci a tracciare il programma col verificarsi di diverse condizioni: per esempio da un certo indirizzo di memoria in poi o a partire dalla transizione di un particolare segnale.

RICERCA DI GUASTI IN UN MICROCALCOLATORE

Metodo di sostituzione

Un metodo di ricerca dei guasti che fa risparmiare molto tempo ed è utilizzato da molti proprietari di piccoli microcomputer (come i personal computer) è quello che consiste nel sostituire sottosistemi sicuramente funzionanti finché il sistema non riprende a funzionare correttamente. Questo metodo viene utilizzato anche per far funzionare per la prima volta un computer nuovo. Supponiamo, per esempio, di avere appena assemblato un nuovo computer con un particolare tipo di sistema operativo, ma che esso non funzioni correttamente. Supponiamo anche di avere un amico con un sistema identico ma funzionante che ce lo presti in modo da permetterci di cominciare a sostituire i pezzi di questo sistema, uno alla volta, nel nostro sistema finché non inizia a funzionare. Questo metodo consente di controllare individualmente vari pezzi. Poiché i collegamenti dei cavi e quelli delle schede di circuiti stampati sono la causa più comune del mancato funzionamento di un sistema, è consigliabile provare a muovere i cavi e i connettori con cautela per assicurarsi che il contatto sia in buone condizioni.

In alcuni sistemi il sistema operativo deve essere modificato perché si adatti alle caratteristiche del disk drive o alle locazioni di I/O utilizzate. Una volta che il sistema operativo per il sistema sotto test è corretto e il guasto è stato isolato ad un dato sottosistema, per determinare la causa del problema è possibile utilizzare alcune delle tecniche trattate nei capitoli 6, 7 e 8. Dopo i problemi di collegamento la causa più probabile del mancato funzionamento di un sistema è il guasto di qualche circuito integrato. È invece molto meno probabile che a causare il malfunzionamento del sistema siano i componenti discreti, resistori o condensatori, presenti sulle schede del circuito.

Test basato sul computer

Il metodo di sostituzione viene utilizzato anche nei piccoli laboratori di riparazione: comunque i laboratori autorizzati dalle varie case costruttrici usano una strumentazione di test più sofisticata e basata sul computer per gestire il sistema e localizzare il guasto in un home computer o in un altro piccolo computer il più rapidamente possibile. Quest'ultimo metodo viene utilizzato anche nelle fabbriche per ricercare i guasti nei sistemi appena costruiti e che non funzionano correttamente.

Quando si sta verificando il funzionamento di un computer, la strumentazione di test del computer sostituisce la CPU del sistema (microprocessore), e prende il controllo dei bus per svolgere le sequenze prescelte di test.

Il test basato sul computer è più veloce e permette di individuare la causa del guasto con più probabilità che non attraverso i semplici test basati sulle misurazioni di tensione e di continuità e di quelli che utilizzano una sonda logica. Generalmente lo strumento di test del computer sostituisce il microprocessore nel sistema sotto test sia fisicamente che elettricamente. Nella sostituzione fisica il microprocessore viene rimosso dal suo zoccolo e sostituito con un connettore collegato allo strumento di test. Nella sostituzione elettrica, invece, il microprocessore viene lasciato al suo posto e lo strumento di test genera un segnale HOLD verso il microprocessore per renderlo inattivo. Questo gli impedisce di mettere segnali sul bus dati, sul bus indirizzi e sul bus di controllo del sistema sotto test. Lo strumento di prova, quindi, prende il controllo di questi bus in modo da svolgere una sequenza prescritta di prove sul sistema sotto test.

Spesso, prima che un sistema sia provato in questo modo, viene controllato il funzionamento del microprocessore sostituendolo con un microprocessore sicuramente ben funzionante. Questo tipo di approccio viene utilizzato solo se il processore è inserito in uno zoccolo e non è collegato o saldato sul circuito stampato. Una volta accertato che il processore funziona correttamente, il guasto deve trovarsi nei sottosistemi di I/O o di memoria. Questi sottosistemi possono essere sostituiti da un gran numero di singoli circuiti anche in un personal computer relativamente poco costoso. Inoltre il funzionamento dei circuiti di ingresso e di uscita varia considerevolmente a seconda delle funzioni che questi stanno compiendo.

Test delle memorie

I dispositivi di memoria possono essere controllati automaticamente verificando tutte le loro locazioni per vedere se funzionano correttamente. Le EPROM e le ROM vengono controllate per vedere se i loro dati sono accurati mentre le RAM vengono controllate anche per verificare la loro capacità nel ricevere e nel memorizzare uno 0 o un 1 in ciascuna locazione.

In generale, per provare il funzionamento di una memoria computerizzata si deve attivare ogni sottosistema in condizioni note in modo tale da determinare se le sue risposte sono, o meno, normali. Alcune forme di questo tipo di test sono già state discusse nel capitolo 6 a proposito delle prove di funzionamento della memoria. Le memorie EPROM e ROM vengono controllate leggendo tutte le loro locazioni e confrontando ciascun byte con un listato o una tabella di byte di riferimento che riporta la sequenza del codice che si sarebbe dovuto avere. Tenendo conto delle deviazioni rispetto alla corretta sequenza del codice, se ve ne sono, il calcolatore di prova o il programma autodiagnostico all'interno di un personal computer può stampare un messaggio che indica la causa più probabile del guasto nel sistema.

Le memorie RAM possono essere controllate analogamente verificando la loro capacità di memorizzare e di restituire sia 0 che 1 in ogni posizione di bit. Questo tipo di test software è fattibile su memorie piccole (con meno di 64.000 byte). Memorie più estese possono richiedere tempi di prova lunghi, ma ciò può anche non sollevare problemi in un laboratorio in quanto si tratta di tempo macchina e non di ore di lavoro di un tecnico. Come si è già detto (vedi *figura 9.1*), le memorie più grandi possono essere costruite in modo tale da comprendere i circuiti di identificazione e di correzione automatica degli errori.

Test del video

Il controllo dei dispositivi di ingresso e di uscita come i terminali video CRT, le stam-

Un programma per verificare il funzionamento di un terminale video è essenzialmente lo stesso di un generico test di memoria, ad eccezione del fatto che vengono utilizzati particolari caratteri (sottolineature e spaziature) per indicare visivamente che tutti i circuiti stanno funzionando correttamente.

panti e le tastiere è simile ai procedimenti trattati nel capitolo 7. Il terminale CRT è composto da una RAM di display, da un circuito controllore del video e da una ROM di visualizzazione del carattere. L'intera RAM può essere controllata in maniera molto simile a quella con cui si controlla una normale memoria di immagazzinamento dati. Per esempio si potrebbe scrivere un programma diagnostico che sottolinei tutte le posizioni del video inviando il codice ASCII 5FH a ciascuna locazione della RAM del display. Uno schermo pieno di sottolineature indica che tutte le locazioni stanno lavorando per quel carattere. Quindi può essere scritto in tutte le locazioni RAM uno spazio (codice ASCII 20H). Se lo schermo risulta tutto nero, la prova è positiva. Questo approccio ha il vantaggio di controllare velocemente il funzionamento della RAM del display, dei circuiti del controllore del display e della ROM dei caratteri, tutti contemporaneamente e con un'indicazione visiva.
I codici utilizzati per le prove del display (tutti spazi o sottolineature) hanno anche il vantaggio che viene provata la capacità della RAM di immagazzinare 0 ed 1, in quanto i due codici utilizzati per provare il sistema sono:

0010 0000 (spazio)
0101 1111 (sottolineatura)

Perciò ogni locazione di memoria del codice ASCII a 7 bit nella RAM del display viene controllata per vedere se può immagazzinare e restituire uno 0 e un 1. Ogni errore nella memorizzazione e nella restituzione del codice binario apparirà sullo schermo CRT come un carattere su uno schermo che dovrebbe essere completamente vuoto oppure come un carattere diverso dalla sottolineatura. Questi caratteri errati appariranno, e forse cambieranno, se la memoria non è in grado di mantenere i suoi segnali al livello 1. Questo è il guasto più comune dei circuiti di una RAM dinamica che vengono utilizzati per le memorie dei display. L'apparizione sul display di codici tutti sbagliati indicherebbe che un certo bit o un dato circuito di memoria è bloccato a 0 o ad 1. Il bit difettoso può essere determinato confrontando il codice ASCII dei caratteri che appaiono in quel momento sullo schermo con il codice ASCII del carattere desiderato.

Test del disk drive

Gli altri principali sottosistemi vengono provati in maniera simile. Per esempio, è possibile scrivere in un disk drive con una serie di segnali noti per poi leggere e confrontare questa serie con quella memorizzata in modo tale da individuare eventuali errori. Gli errori di memorizzazione di un disco si ritrovano anche come byte inesatti per l'individuazione dell'errore (il segnale CRC controlla il byte alla fine della registrazione di ciascun dato) nel dato registrato o letto. Questo errore viene solitamente indicato da un messaggio di errore sullo schermo durante le operazioni di routine. Un disk drive può non funzionare perché è guasto uno dei suoi controllori, oppure perché ci sono problemi nel cablaggio o nel cavo che collega il controllore con il disk drive e le schede del processore. I malfunzionamenti possono essere causati anche dalle testine magnetiche di registrazione e di lettura difettose, oppure dal disco stesso. Se tutte le registrazioni sono sbagliate, il problema riguarda le testine, il cablaggio o il controllore del disco. Se solo alcune registrazioni sono sbagliate, il problema riguarda il disco o un contatto intermittente nel cablaggio. La sostituzione del disco può velocemente e semplicemente confermare o eliminare questa possibilità. Una volta individuato uno specifico malfunzionamento, queste possibilità di guasto possono essere scritte sullo schermo CRT o su un messaggio stampato dalla stampante.

Test di interfaccia

Uno strumento programmabile basato su di un computer può controllare completamente il funzionamento dei circuiti d'ingresso e di uscita di un sistema, applicando configurazioni note agli ingressi e confrontando le risposte con configurazioni note in uscita.

Lo strumento di test può essere predisposto in modo tale da provare i dispositivi di ingresso e di uscita, come i circuiti di interfaccia di dati seriali, fornendo loro delle sequenze note di dati in ingresso. Il funzionamento del circuito può essere determinato controllando se il circuito di interfaccia ha ricevuto i dati esatti. Poiché lo strumento di prova controlla i dati in ingresso e contemporaneamente registra le risposte del circuito di ingresso, è possibile per esso determinare quale sia il guasto più probabile. Solitamente lo strumento di prova è programmabile ed ha un gran numero di connettori o di sonde che possono essere collegati col sistema sotto test. Esso esaminerà tutte le sequenze di risposta a dei segnali di ingresso per determinare quali dispositivi di ingresso e ingressi stiano funzionando correttamente e quali invece siano guasti e perché. Fornendo al computer di prova gli appropriati convertitori analogico-digitali, sarà possibile misurare automaticamente anche le quantità analogiche. Certi segnali come i clock e i livelli di tensione dell'alimentazione possono essere controllati sotto diverse condizioni di test per determinarne i guasti. Generalmente tanto più estensivamente viene provato l'intero sistema del computer con segnali di test e sequenze di segnali ben approntati, tanto più precisi saranno i messaggi diagnostici che verranno generati e mostrati sul display al riparatore. Quest'ultimo, allora, si potrà concentrare su un'area del computer abbastanza circoscritta, per compiere prove dettagliate con un oscilloscopio od un analizzatore logico.

Test della tastiera

Per trovare i guasti in una tastiera essa deve essere controllata secondo una particolare sequenza, cosicché il computer possa verificare il codice corretto.

Come ultimo esempio di prova controllata dal computer di un sottosistema di ingresso verrà considerata la tastiera già trattata nel capitolo 7. Per provare una tastiera il computer dovrebbe far sì che i tasti giusti vengano premuti in una sequenza nota tramite l'uso di una matrice di solenoidi montata sopra la tastiera; oppure un operatore dovrebbe premere i tasti seguendo un certo ordine. Si dovrebbero collegare delle sonde con le righe e le colonne della matrice di interruttori e con le uscite del circuito codificatore e del circuito di scansione. Effettuando poi le chiusure degli interruttori dei tasti seguendo un certa sequenza, il computer può confrontare i codici di tasto, di riga e di colonna reali della matrice con quelli che si dovrebbero osservare con ciascun tasto.

Se i codici di riga e di colonna non corrispondono ai loro valori normali, allora il codice di tasto sarà anch'esso sbagliato. In tal caso, il computer di prova può stampare che si sospetta essere aperto o guasto un tasto di cui si specifica anche la riga e la colonna. Esso potrebbe anche stampare il codice ASCII del carattere che ha ricevuto insieme a quello vero in modo da aiutare il riparatore nel determinare quali fili o quali tasti siano guasti. Se i codici di riga e di colonna sono corretti per tutte le chiusure dei tasti, ma il codice di tasto è sbagliato, il computer di prova può stampare il messaggio che c'è un guasto nel circuito codificatore del tasto o nel suo cablaggio. Se la parte diagnostica del computer è sufficientemente complessa, è possibile che i messaggi di errore specifichino quali cavi o quali circuiti sono guasti e perché. Se si dovesse provare la tastiera usata come esempio nel capitolo 7, il computer determinerebbe che tutti i tasti sulla parte sinistra della tastiera producono codici ASCII sbagliati, mentre i codici di riga e di colonna sono giusti. Il programma di test potrebbe poi determinare che la matrice dei tasti funziona correttamente ma che è errato il processo di codificazione. Inoltre, confrontando i codici ASCII ricevuti con quelli corretti, il programma di test potrebbe decidere che la linea D_3 è sempre bloccata ad 1, pervenendo quindi alla stessa conclusione a cui eravamo arrivati noi analizzando i dati. Il vantaggio del test col computer è che l'analisi dei codici ASCII e la prova dei tasti vengono compiute automaticamente in un periodo di tempo molto breve con un messaggio di errore che indica che la linea D_3 uscente dal decodificatore è bloccata ad 1. Perciò il riparatore può controllare immediatamente il circuito del decodificatore e il cablaggio tra il circuito e il processore.

Con un numero sufficiente di sonde di ingresso, il computer di prova potrebbe perfino determinare se la condizione di blocco ad 1 si trova nel circuito del decodificatore, nel collegamento con il circuito stampato o nel cablaggio di ritorno al processore attraverso qualche buffer di ingresso. Il calcolatore di prova potrebbe quindi stampare il seguente messaggio: "Il circuito del decodificatore deve essere sostituito" oppure "Si controlli il collegamento del piedino n... del decodificatore" oppure "C'è un'apertura nel cablaggio tra il piedino n... del circuito decodificatore e il piedino n... del buffer d'ingresso" e così via. Ne risulta una veloce identificazione del guasto che lascia al riparatore solo il semplice compito di riparare il cablaggio o un connettore oppure di sostituire un circuito.

STRUMENTAZIONE DI RICERCA DEI GUASTI NEI CALCOLATORI DI USO GENERALE

Un approccio strettamente legato al test basato sul computer è quello di utilizzare uno strumento di prova programmabile progettato appositamente per ricercare i guasti nei computer. Un esempio di questo tipo di strumento è il Fluke 9010A Microsystem Troubleshooter mostrato in *figura 9.17*. Questo strumento può essere utilizzato come un analizzatore di firma, ma è progettato come uno strumento di prova di uso generale basato sul calcolatore, che può essere usato con un qualsiasi microcomputer (una volta installata un'appropriata sonda del microprocessore). Esso è progettato in modo tale che lo si possa adattare ad ogni sistema sotto test (entro limiti ragionevoli). Questo approccio è differente da quello trattato in precedenza, perché prima si era supposto di utilizzare un sistema di test per calcolatore progettato e realizzato appositamente per un particolare tipo di computer.

Tra le caratteristiche principali del 9010A è inclusa una sonda che sostituisce il microprocessore stesso nel sistema sotto test. Ciò permette al 9010A di attivare il resto del sistema. Il 9010A è un dispositivo programmabile attraverso un interfaccia se-

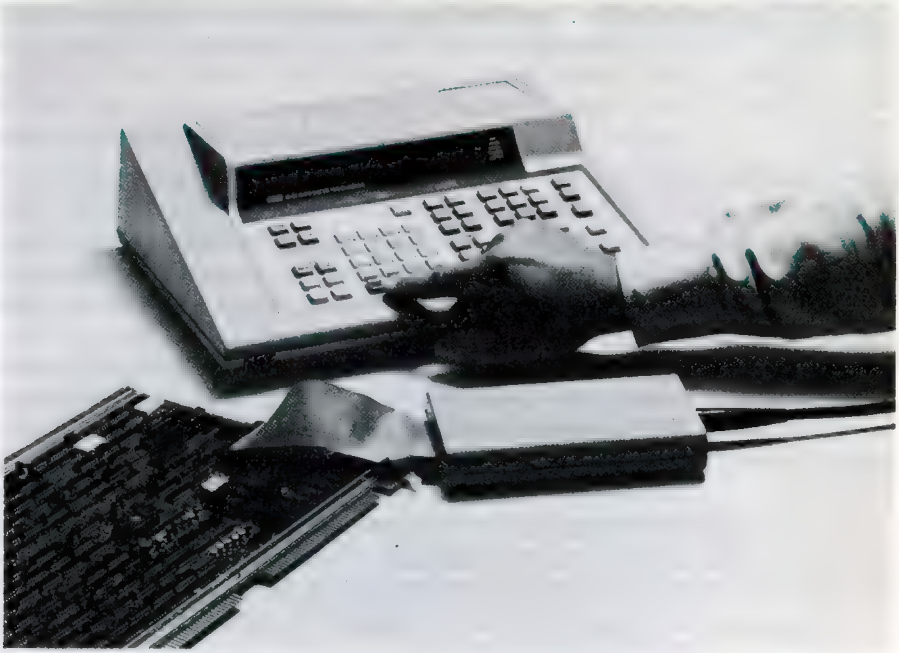


Figura 9.17. Microsystem Troubleshooter *(Per gentile concessione della John Fluke, Mfg. Co.)*

Con un sistema di ricerca dei guasti programmabile, utilizzato nella ricerca dei guasti in computer, il riparatore fa funzionare il sistema sostituendo il processore del computer con un processore del sistema di ricerca guasti.

riale RS232, un nastro a cassetta o la sua stessa tastiera. Questa possibilità di programmazione permette all'operatore di dare al 9010A istruzioni, cioè di indicargli quale indirizzo corrisponde ad un certo tipo di dispositivo (RAM, ROM, certi tipi di I/O, ecc.). Esso permette inoltre all'operatore di programmare quali test devono essere compiuti su ciascuno di questi dispositivi e secondo quale sequenza. Lo strumento potrà essere programmato in modo tale da stampare messaggi diagnostici o di stato e istruzioni per l'operatore. Un esempio di istruzione potrebbe essere la richiesta di sondare un certo piedino o un circuito integrato del sistema sotto test. L'operatore può sondare manualmente certe parti del sistema mentre lo strumento di prova registra la reazione del sistema. Si tratta della combinazione molto efficace di un veloce strumento di test computerizzato e di un operatore che permette di localizzare velocemente i guasti nei sistemi complessi.

Il metodo di "autoprogrammazione"

Una delle caratteristiche più importanti di uno strumento come il 9010A è la sua capacità di autoprogrammarsi. Questo può ridurre di molto il tempo di programmazione dell'operatore e fornire una procedura di test relativamente completa per un qualsiasi sistema dato. Lo strumento entra in questo modo di funzionamento "autoprogrammato" quando viene premuto il comando LEARN sulla console dello strumento dopo che le sonde del microprocessore sono state collegate ad un sistema sicuramente funzionante. Lo strumento di test passa quindi attraverso tutti i possibili indirizzi di memoria e di I/O. Ciò viene fatto per determinare se c'è un componente che risponde a ciascun indirizzo e per vedere come risponde. Questo permette al 9010A

Nella modalità autodiagnostica, il sistema di ricerca dei guasti computerizzato registra le risposte appropriate di un sistema sicuramente funzionante. Esso utilizza queste risposte, immagazzinate nella memoria, per confrontarle con quelle del sistema sotto test, in modo da determinare ogni discrepanza.

di acquisire completamente le caratteristiche del sistema funzionante. Esso può determinare quali indirizzi di memoria non vengono utilizzati, quali vengono utilizzati per la memoria del programma e quale programma è immagazzinato in quella memoria. Esso può anche determinare quali locazioni sono RAM e come ciascun indirizzo di I/O risponde a un segnale noto. Una volta stabiliti gli indirizzi di memoria e di I/O (cioè stabilita la cosiddetta mappatura) e come ciascun indirizzo risponde, il 9010A è pronto per attivare tutte queste locazioni. Attivando questi indirizzi in un sistema sotto test lo strumento di prova può controllare le loro risposte e confrontarle con quelle corrette. In questo modo possono essere isolati velocemente e senza un eccessivo sforzo di programmazione da parte dell'operatore i maggiori guasti (problemi dei bus o dei componenti).

Per controllare le più sofisticate parti di I/O del sistema, l'operatore avrà bisogno di programmare speciali sequenze di test per lavorare con lo strumento di prova utilizzando il metodo di autoprogrammazione. Ciò permetterà allo strumento di verificare il funzionamento dei dispositivi seriali di interfaccia e delle unità di controllo video e disco. Lo strumento può, per esempio, essere programmato in modo tale da fermarsi dopo che ha provato tutti i componenti di memoria, i semplici componenti di I/O e i componenti dei bus. Quindi può dare istruzioni all'operatore perché sondi l'uscita di un chip di interfaccia seriale. In seguito, lo strumento può mandare una serie di caratteri ASCII a quel dispositivo mentre registra il dato seriale che risulta. Se il dato seriale è corretto per i codici inviati, è stato verificato il funzionamento della parte d'uscita dell'interfaccia. Lo strumento può quindi dare all'operatore l'istruzione di porre la sonda sull'ingresso dell'interfacca seriale e di inviare un'altra serie di dati seriali. Esso invierà quindi i codici forniti dall'interfaccia al bus dati in parallelo del sistema per vedere se i codici vengono ricevuti correttamente. Altri segnali del componente di interfaccia seriale, come il $\overline{\text{CTS}}$ o il $\overline{\text{DTR}}$, possono essere provati analogamente come anche le singole linee dei circuiti di interfaccia in parallelo. Questi test sono abbastanza lenti, dato che l'operatore deve trovare il punto adatto per il test, ma costituiscono quasi l'unico modo possibile per provare esaurientemente tali componenti. I convertitori analogico-digitali o digitali-analogici possono essere provati in maniera simile utilizzando ingressi analogici noti e misurando l'uscita analogica generata dal 9010A.

Certamente l'uso di un tale strumento permette di individuare quasi il 100% dei guasti in tempi che vanno da un minimo di molto meno di un'ora ad un massimo di qualche ora. Questo va confrontato con le percentuali di individuazione dei guasti molto più basse che un riparatore può ottenere utilizzando solo un oscilloscopio o un voltmetro ed impiegando qualche giornata di lavoro.

TECNICHE FUT

Molte delle tecniche presentate in questo capitolo indicano qual è il futuro della ricerca guasti nei sistemi digitali. Nei sistemi più grandi la tendenza è quella di includere in essi sempre più software ed un numero sempre maggiore di circuiti che possano autonomamente individuare gli errori e diagnosticare i guasti. Queste aggiunte aumenteranno il costo del sistema, ma renderanno molto più semplice la sua manutenzione e la sua riparazione.

Sistema diagnostico separato

Sono stati fatti tentativi per costruire un sistema in parallelo all'interno di un sistema digitale solo per controllare il funzionamento del sistema originale ed individuare un qualsiasi malfunzionamento. In questo campo devono ancora essere fatti significativi progressi.

All'interno di un dato sistema potrebbero esserci due sistemi digitali in parallelo: uno, che si usa normalmente, e l'altro che è un sistema di monitoraggio per controllare il funzionamento ed individuare gli errori. Il sistema di monitoraggio individuerà un malfunzionamento e segnalerà all'operatore che cosa è necessario fare per correggerlo.

Naturalmente, l'affidabilità dei dati provenienti dal sistema di monitoraggio dipende dal corretto funzionamento dello stesso. Se dovesse verificarsi un guasto nel processore centrale, nell'alimentazione principale o nello stesso sistema diagnostico, il software diagnostico non sarebbe in grado di svolgere i suoi test. Di conseguenza è possibile che si verifichi un guasto che non può essere rilevato o che il sistema diagnostico non è in grado di isolare.

Se la parte diagnostica del sistema controlla periodicamente le tensioni nei vari punti di alimentazione, può individuare i guasti dell'alimentazione stessa notando un rapido calo in una o più delle tensioni di alimentazione misurate. Il sistema diagnostico dovrebbe però avere ancora abbastanza energia immagazzinata, o potrebbe attivare l'alimentazione di emergenza fornita da una batteria, per trasmettere sulla stampante un messaggio di errore, come per esempio:

```
L'ALIMENTAZIONE PRINCIPALE È GUASTA NELLA SEZIONE A +5 VOLT; È STATA INSE-
RITA LA BATTERIA AUSILIARIA.
COMPONENTI SOSPETTI: DIODO D2 E CONDENSATORE DEL FILTRO C4.
```

Nel caso di guasti veramente critici, come ad esempio l'improvviso mancato funzionamento del microprocessore, il sistema non può fornire alcuna diagnosi. Un approccio che può essere utilizzato per prevenire questo tipo di guasto è quello di inserire dei processori di riserva che controllino gli altri processori del sistema. Se un processore smettesse quindi di funzionare correttamente, gli altri processori potrebbero decidere di escluderlo elettricamente dal circuito e di assegnare i sùoi compiti ad uno dei processori di riserva. Contemporaneamente potrebbe essere stampato un messaggio del tipo:

```
SOSTITUIRE PROCESSORE n...
```

mentre il sistema continua a funzionare normalmente.

Se il sistema non ha tali caratteristiche autodiagnostiche o non ha sufficienti capacità di prova per controllare e localizzare tutti gli errori, ci sarà sempre la strumentazione di test basata su computer che controllerà automaticamente, in gran parte, il computer malfunzionante finché ne determinerà il guasto. Il riparatore dovrà possedere ancora la capacità di analizzare e l'abilità di provare il funzionamento dei sistemi digitali al fine di riparare il guasto sospetto indicato dai sistemi diagnostici. Nei sistemi semplici, le caratteristiche diagnostiche saranno limitate e si dovranno utilizzare le tecniche descritte in questo libro per localizzare il guasto nella vecchia maniera e attraverso un duro lavoro.

CHE COSA ABBIAMO IMPARATO?

1. Molti sistemi di memoria hanno circuiti di individuazione automatica degli errori (detti generatori di parità) e di correzione degli errori già inclusi nel progetto.
2. Per isolare un guasto possono essere necessari molti test differenti eseguiti con differenti circuiti e strumentazioni di prova.
3. Le relazioni di temporizzazione sono più critiche nei sistemi più complessi e richiedono l'uso di oscilloscopi a più canali o di analizzatori logici per isolare il guasto.
4. Un analizzatore di firma può visualizzare sul display sequenze di bit di dati e caratteristiche del circuito, ma non può mostrare le relazione di temporizzazione, né identificare i glitch.
5. Anche con i test controllati dal computer il riparatore deve spesso ricorrere alle operazioni manuali per aiutare il computer ad isolare il guasto.
6. La quantità di dati visualizzati sul display da un analizzatore logico è il prodotto della memoria dell'analizzatore per il tempo di campionamento.

Quiz del Capitolo 9

1. Se i dispositivi di memoria di *figura 9.1* avessero 256K locazioni (un bit per locazione), come cambierebbe il progetto?
   a. Non sarebbe necessario alcun cambiamento.
   b. Non andrebbe bene la stessa struttura.
   c. L'unità di controllo del ripristino e il generatore del segnale $\overline{RAS}$ dovrebbero essere cambiati.
   d. Si dovrebbe cambiare la correzione degli errori.

2. Nelle ipotesi della domanda 1, quante linee di indirizzo dovrebbe fornire il processore per la struttura di *figura 9.1*?
   a. 16
   b. 18
   c. 20
   d. 22

3. Nell'ipotesi della domanda 1, di quanti megabyte dovrebbe essere la memoria?
   a. 0,256
   b. 2,048
   c. 4,096
   d. 8,192

4. Quanti piedini di indirizzo dovrebbe avere ciascun circuito?
   a. 9
   b. 16
   c. 18
   d. 20

5. Sempre nel caso della prima domanda, quante delle linee di indirizzo del processore verrebbero inviate al multiplexer di indirizzo del circuito di memoria? (Il multiplexer è il dispositivo che funge da interfaccia tra le linee di indirizzo di ordine basso e gli ingressi di indirizzo sui singoli dispositivi di memoria.)
   a. 9
   b. 16
   c. 18
   d. 20

6. Se in *figura 9.1* il codice dati inviato ad una locazione è 93CAH, quale sarà il valore memorizzato nei bit di parità compresi tra P_1 e P_6?
   a. 011110
   b. 100010
   c. 101110
   d. 010001

7. Se nella domanda precedente i bit di parità fossero 111011, che cosa si dovrebbe fare?
   a. Il bit 11 dovrebbe essere corretto in 0.
   b. Il bit 11 dovrebbe essere corretto in 1.
   c. Il bit 4 dovrebbe essere corretto in 0.
   d. Il bit 4 dovrebbe essere corretto in 1.

8. Sempre nel caso della domanda 6, se il codice di parità letto dalla locazione è 111111, che cosa si dovrà fare?
   a. Indicare un errore di parità nel bit P_1
   b. Indicare un errore di parità nel bit P_6
   c. Indicare un errore di parità sia nel bit P_1 che nel bit P_6
   d. Correggere il bit 11.

9. Lo scopo dell'analizzatore logico è di:
   a. Verificare il funzionamento logico delle porte di un sistema.
   b. Campionare e visualizzare sul display i segnali di un sistema.
   c. Analizzare le operazioni logiche funzionali di un sistema.

10. Se un analizzatore logico utilizza un clock con una frequenza di 50 megahertz e una memoria da 4000 parole, che tempo rappresenta, in microsecondi, l'ampiezza orizzontale del display?
    a. 20
    b. 40
    c. 80
    d. 800

11. In *figura 9.12* il segnale sulla linea A_0 indica:
    a. Che si è verificato un picco di rumore sulla linea A_0.
    b. Che il contatore di riga è stato incrementato due volte.
    c. Che c'è un problema col generatore del segnale $\overline{RAS}$.
    d. Che c'è un problema con il generatore del clock di ripristino.
    e. Nessuna delle risposte precedenti.
    f. Le risposte b e d.

12. Un glitch del tipo mostrato in *figura 9.13* agisce:
    a. Come un secondo impulso di clock.
    b. In modo tale da provocare un secondo segnale $\overline{RAS}$.
    c. In modo tale da provocare un errore sul segnale A_1
    d. In modo tale da mantenere la linea A_2 a 0.

13. Un glitch è:
    a. Una sequenza momentanea di segnali da 0 a 1 a 0.
    b. Una sequenza momentanea di segnali da 1 a 0 a 1.

c. Un impulso di rumore.
d. Una qualunque delle risposte precedenti.

14. Un display a soglia doppia in un analizzatore logico fornisce informazioni riguardo:
a. Al verificarsi dei glitch.
b. Al tempo di salita.
c. Al tempo di discesa.
d. Ai ritardi di propagazione.
e. A tutti i casi precedenti.
f. Ai casi b e c precedenti.

15. Se un analizzatore logico è regolato per far apparire sul display le informazioni in forma esadecimale, fornendo un totale di sei cifre esadecimali, quante sonde di segnale e quanti canali dell'analizzatore occorrono?
a. 6
b. 18
c. 24
d. 30

16. La sequenza di segnali mostrata in *figura 9.15* indica:
a. Le stesse informazioni di base delle forme d'onda di *figura 9.12* per il funzionamento delle linee di indirizzo.
b. La possibile presenza di glitch casuali.
c. L'occasionale incremento doppio del contatore di riga.
d. Uno qualunque dei casi precedenti.

17. Se il display di *figura 9.15* avesse continuato a visualizzare tutti 00, il problema più probabile sarebbe (*figura 6.9*):
a. Un tempificatore di ripristino guasto.
b. Un contatore guasto.
c. Un'apertura sulla linea RFRQ.
d. Un guasto sulla linea RFGR.
e. Una qualsiasi delle risposte precedenti.

18. Se il display di *figura 9.15* presentasse tutti 00 oppure delle cifre esadecimali a caso sull'uscita del multiplexer di indirizzi mentre le uscite del contatore di riga sono corrette (00, 01, 02, 03, ecc.), si tratterebbe probabilmente di (*figura 6.9*):
a. Un guasto nel multiplexer o nelle sue linee di controllo.
b. Un contatore guasto.
c. Un temporizzatore di ripristino guasto.
d. Un segnale RFRQ sbagliato.
e. Uno qualsiasi dei casi precedenti.

19. Alcuni analizzatori logici possono essere programmati in modo tale da visualizzare sul display istruzioni in linguaggio assemblativo proprio nel momento stesso in cui vengono eseguite.
a. Vero.
b. Falso.

20. L'approccio generalmente usato dalla strumentazione automatica di test per un calcolatore consiste nel:
a. Far apparire sul display i segnali dell'analizzatore logico.
b. Sostituire i sottosistemi che si sospettano guasti con altri sicuramente funzionanti.
c. Fornire un segnale standard ad ogni parte del calcolatore, verificarne la risposta e ripetere questa operazione per tutte le parti del sistema sotto test.

Appendice

FUNZIONAMENTO DELL'OSCILLOSCOPIO

L'oscilloscopio permette di visualizzare su di uno schermo un segnale al variare del tempo tramite un grafico dell'ampiezza del segnale sotto test in funzione del tempo. Come schermo viene utilizzato un tubo a raggi catodici (CRT) molto simile a quello dei televisori. Il CRT è composto da un cannone di elettroni, da placche di deflessione e da uno schermo rivestito di fosfori e interamente contenuto in un tubo di vetro sotto vuoto, come mostrato in *figura A.1*. Il cannone di elettroni produce un sottile fascio di elettroni che viene "sparato" verso la superficie del CRT. Quando gli elettroni colpiscono il fosforo sulla superficie del CRT viene prodotto un punto (spot) di luce. Il fosforo emette una luce di un determinato colore: negli oscilloscopi essa è generalmente verde.

Lo spot può essere ridotto ad un piccolo punto tramite il controllo del fuoco posto sul pannello frontale dell'oscilloscopio. Il controllo dell'astigmatismo, che si trova su alcuni oscilloscopi, ritocca lo spot in modo che abbia una forma quasi perfettamente circolare. Il controllo di luminosità varia la luminosità o l'intensità dello spot. Questo controllo di solito è tale che la luminosità dello spot può variare tra la eliminazione totale e una condizione di elevata luminosità.

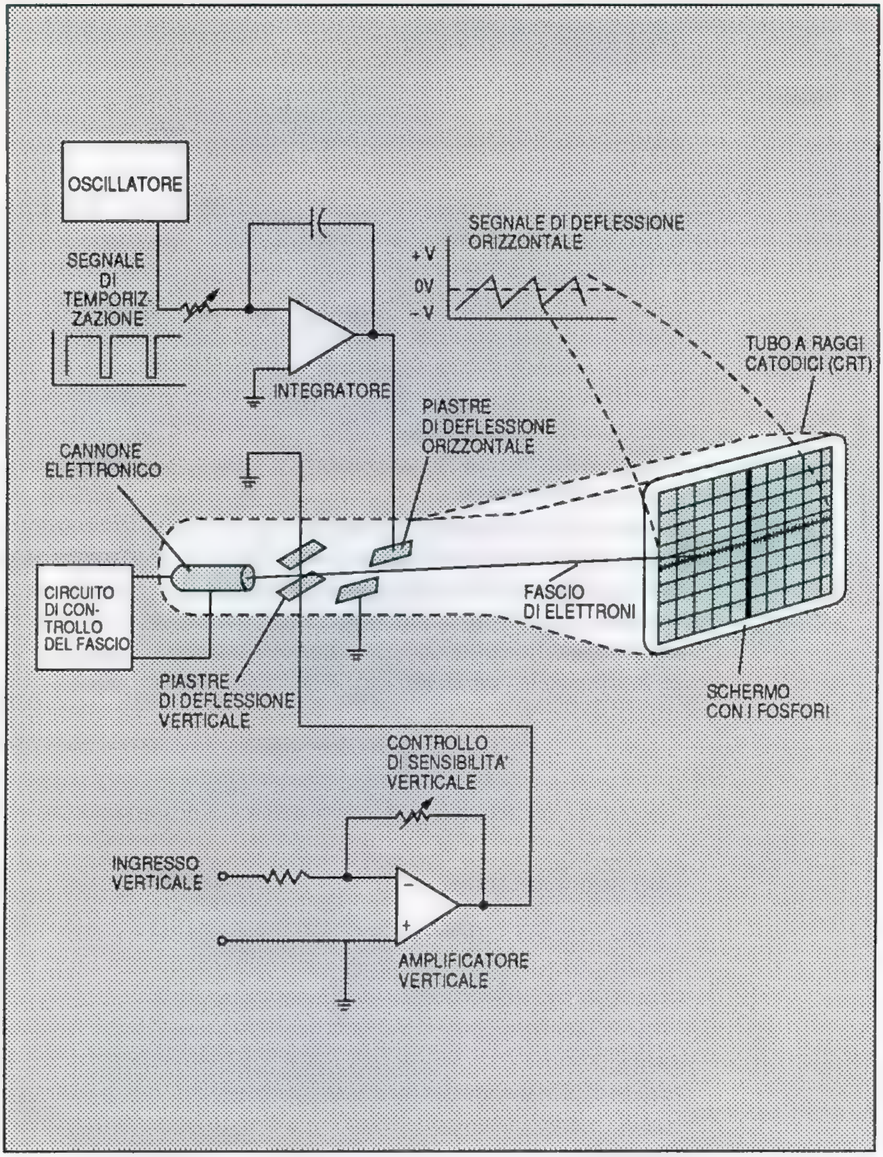
> *ATTENZIONE: La luminosità non deve essere lasciata ad un livello alto quando sullo schermo è presente un solo spot; così facendo si può danneggiare lo schermo.*

Deflessione del fascio

I segnali applicati alle placche di deflessione fanno muovere il punto luminoso verso l'alto, verso il basso e in orizzontale sullo schermo del CRT. Il segnale che fa muovere il punto luminoso attraverso lo schermo, solitamente da sinistra verso destra, viene chiamato segnale di deflessione orizzontale. Questo segnale ha una forma come quella mostrata nelle *figure A.1* ed *A.2*: perciò viene chiamato forma d'onda a dente di sega. La sua tensione è direttamente proporzionale al tempo: cioè, all'aumentare del tempo l'ampiezza della forma d'onda aumenta in maniera direttamente proporzionale.

In *figura A.1* va notato che la placca di deflessione orizzontale sinistra è collegata a massa mentre quella di destra è collegata alla tensione di deflessione. Quando la tensione di deflessione è negativa, il punto è costretto a muoversi nella parte sinistra dello schermo. All'aumentare della tensione il punto si muove verso la parte destra. Il punto si trova invece esattamente al centro dello schermo quando la tensione è 0 e all'estrema destra dello schermo quando la tensione è massima. La velocità con cui il punto si muove attraverso lo schermo dipende dalla frequenza della tensione di deflessione orizzontale. A frequenze molto basse è possibile vedere realmente il punto muoversi. Comunque, a frequenze più elevate, il punto si muove così velocemen-

Figura A.1. Schema di funzionamento di un oscilloscopio


OSCILLATORE
SEGNALE DI TEMPORIZZAZIONE
INTEGRATORE
SEGNALE DI DEFLESSIONE ORIZZONTALE
+V
0V
-V
TUBO A RAGGI CATODICI (CRT)
PIASTRE DI DEFLESSIONE ORIZZONTALE
CANNONE ELETTRONICO
CIRCUITO DI CONTROLLO DEL FASCIO
FASCIO DI ELETTRONI
PIASTRE DI DEFLESSIONE VERTICALE
SCHERMO CON I FOSFORI
CONTROLLO DI SENSIBILITA' VERTICALE
INGRESSO VERTICALE
−
+
AMPLIFICATORE VERTICALE

te che è possibile vedere solamente una linea continua, chiamata traccia, attraverso lo schermo. La velocità con cui il punto si muove viene detta "sweep rate" e viene misurata in unità di tempo in funzione della distanza. Uno o più comandi di controllo permettono all'operatore di fissare la sweep rate.

Lettura dello schermo

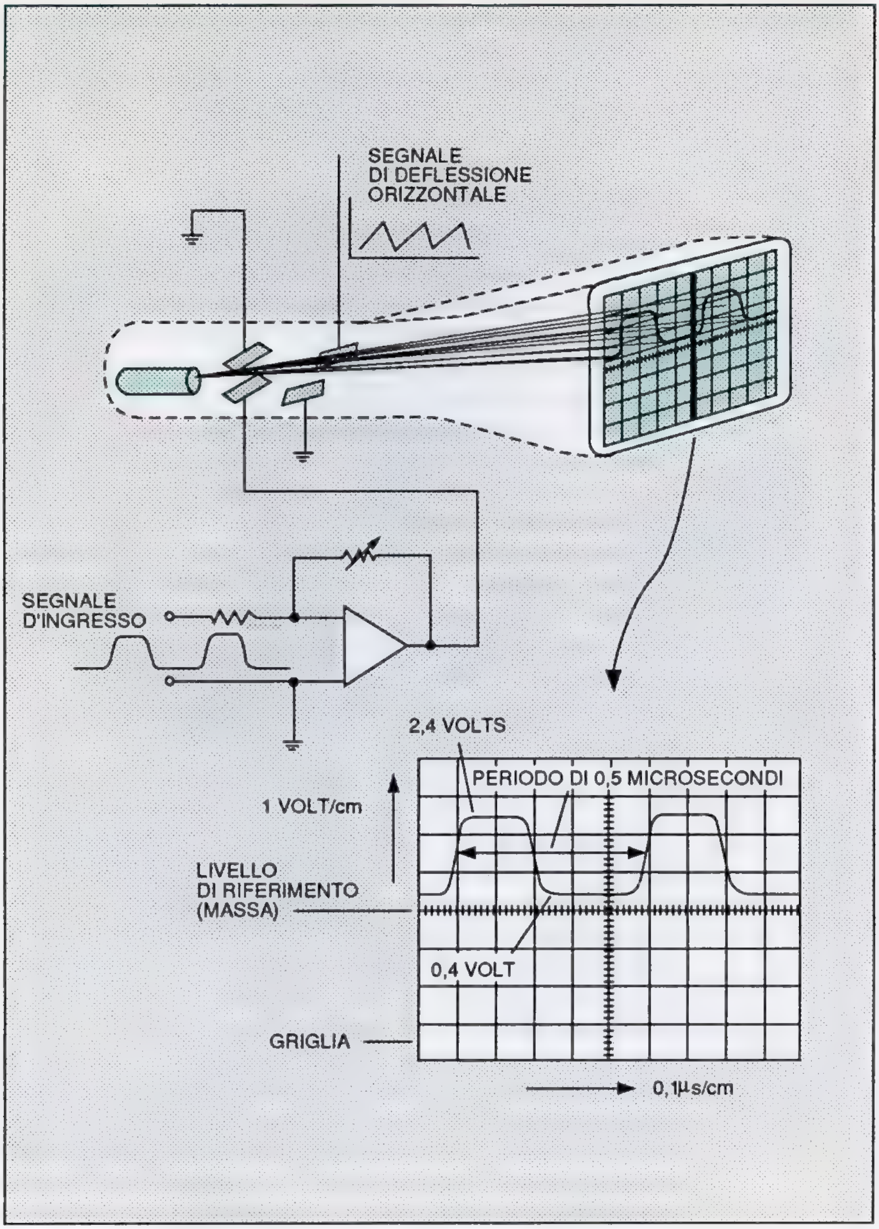
Sullo schermo del CRT è riportata una quadrettatura che lo rende simile ad un foglio di carta millimetrata. Questa quadrettatura viene detta "griglia". Fissando la sweep rate o il controllo tempo/divisione si determina il tempo necessario al punto per attraversare ogni quadretto della griglia. Di solito ogni quadretto ha il lato di un centimetro, che è diviso in cinque parti da quattro piccoli trattini. Perciò se la scala dei tempi della sweep è stata calibrata in funzione di un certo tempo per centimetro, il periodo di una forma d'onda visualizzata sullo schermo può essere misurato con precisione.

Il segnale la cui forma d'onda è visualizzata sullo schermo è applicato all'ingresso dell'amplificatore verticale. Il segnale amplificato è applicato alle placche di deflessione verticale del CRT come indicato in *figura A.1*. La tensione applicata a queste placche fa muovere il punto verso l'alto o verso il basso: perciò il punto seguirà l'ampiezza del segnale in ingresso (forma d'onda della tensione). Se non è applicata alcuna tensione di deflessione orizzontale quando un segnale è applicato all'amplificatore verticale, il punto si muove verso l'alto e verso il basso, cosicché si osserva una linea verticale. Questo di solito non è molto utile e quindi, normalmente, la sweep orizzontale viene attivata.

Quando la sweep rate è stata scelta correttamente, viene visualizzata sul display la forma d'onda del segnale presente all'ingresso verticale. Poiché la griglia presenta anche delle suddivisioni verticali, può essere misurata l'ampiezza del segnale se il controllo della sensibilità verticale è calibrato in unità di volt per suddivisione della griglia. Perciò con una sweep rate calibrata in tempo per suddivisione e con un amplificatore verticale calibrato in volt per suddivisione, possono essere misurati il periodo e l'ampiezza di un segnale d'ingresso. Dato che la frequenza può essere determinata come f=1/T, la frequenza di una forma d'onda periodica può essere determinata in base al periodo che abbiamo misurato. (Per quanto riguarda le caratteristiche di un segnale ci si riferisca al capitolo 1.)

Le scale dei tempi vengono generalmente indicate con delle scritte del tipo 0,1 microsecondi/cm, 1 microsecondo/cm, e così via, poste sul pannello. In *figura A.2* è stata scelta la scala di 0,1 microsecondi/cm. Poiché la dimensione orizzonatale della griglia comprende 8 quadretti, il punto si muove attraverso l'intera griglia in 0,8 microsecondi. Dato che ogni quadretto rappresenta 0,1 microsecondi, ciascuna delle cinque suddivisioni rappresenta 0,02 microsecondi. Osservando questo display è facile determinare che il periodo del segnale è 0,5 microsecondi. Perciò la frequenza del segnale è 2 megahertz. Risulta inoltre facile determinare che il segnale è attivo (al livello logico 1) per 0,2 microsecondi e rimane al livello logico 0 per la parte restante del periodo. I tempi di salita e di discesa possono essere stimati nell'ordine di 0,02 microsecondi (20 nanosecondi) o meno, poiché le transizioni dal livello basso al livello alto e dal livello alto al livello basso si verificano in meno di una suddivisione.

Figura A.2. Tracciato del segnale di clock su di un oscilloscopio


SEGNALE
DI DEFLESSIONE
ORIZZONTALE
SEGNALE
D'INGRESSO
2,4 VOLTS
PERIODO DI 0,5 MICROSECONDI
1 VOLT/cm
LIVELLO
DI RIFERIMENTO
(MASSA)
0,4 VOLT
GRIGLIA
0,1µs/cm

Per avere una misurazione più accurata, il controllo tempo/suddivisione deve essere fissato sulla scala di 0,01 microsecondi/cm. Il segnale apparirà così ingrandito di 10 volte in senso orizzontale rispetto a quello mostrato nel display di *figura A.2*.

Uso dell'oscilloscopio

I livelli logici 0 ed 1 del segnale possono essere determinati una volta nota la scala verticale e stabilito un livello a 0 volt (ground). Questa scala viene scelta tramite un controllo di sensibilità verticale calibrato in volt/cm. La posizione del riferimento di massa può essere scelta collegando l'ingresso verticale massa e muovendo la traccia con il controllo di posizione verticale fino a farla coincidere con una delle linee orizzontali della griglia. In *figura A.2* il livello di riferimento era la linea orizzontale centrale. Quindi applicando il segnale da analizzare all'ingresso verticale e la massa del segnale alla massa dell'oscilloscopio, il display del tipo mostrato in *figura A.2* indica i livelli di tensione rispetto al livello di riferimento noto. Fissando la scala del controllo di sensibilità verticale a 1 volt/cm, è possibile determinare la tensione di ciascun punto sulla forma d'onda. In *figura A.2* il livello logico 0 è a circa 0,4 volt (due suddivisioni piccole sopra il livello di riferimento) e il livello logico 1 è a circa 2,4 volt. Perciò il segnale è un tipico segnale TTL in termini di livelli logici 0 ed 1.

Questa trattazione ha presentato solo le funzioni di base di un oscilloscopio. La maggior parte degli oscilloscopi hanno diverse altre funzioni e controlli che li rendono strumenti molto versatili. Dato che queste funzioni e l'uso dei comandi variano da modello a modello, l'operatore dovrebbe consultare il manuale di istruzioni fornito dalla casa costruttrice per avere queste informazioni.

Glossario

Abilitante: Termine dato ad un segnale che permette al circuito di emettere dati.
Abilitazione di uscita (OE): Segnale che, quando è attivo, collega una cella di memoria con le linee di uscita del dispositivo.
Addizionatore: Blocco costitutivo che fornisce una somma e un riporto quando vengono sommati due numeri.
ALU: Unità aritmetico-logica (Arithmetic Logic Unit). Fa parte dell'unità centrale del processore (CPU) che svolge tutte le operazioni matematiche richieste da un programma del computer.
Analizzatore logico: Strumento che campiona le tensioni rilevate e visualizza su uno schermo CRT la sequenza dei livelli del segnale (0 o 1).
Analogico: Il circuito analogico, chiamato anche "circuito lineare", è un circuito che varia alcune proprietà elettriche con continuità e senza brusche variazioni all'interno di un certo intervallo.
Asincrono: Si riferisce a circuiti e operazioni privi di un comune segnale di clock.
Binario: Sistema di numerazione in base 2 che utilizza le cifre 0 e 1. I numeri binari sono comuni nelle operazioni dei computer in quanto la maggior parte dei dispositivi funziona con modalità a due stati (1=acceso, 0=spento).
Buffer a tre stati: Circuito d'uscita che, quando viene disabilitato, scollega le tensioni di uscita dalle linee di uscita.
Bus: Percorso all'interno del computer che è comune a diversi dispositivi (come il bus indirizzi o il bus dati). Quando un indirizzo viene posto sul bus indirizzi, esso raggiunge contemporaneamente tutti i dispositivi sul bus (come la memoria o un dispositivo periferico), ma solo il dispositivo il cui indirizzo corrisponde a quello presente sul bus risponderà.
Campionamento: Termine usato per descrivere il controllo del funzionamento di un segnale di un circuito o di un sistema effettuato in maniera casuale. Con un sufficiente numero di campioni (basato sulla teoria di linearità) è possibile determinare lo stato operativo di un dato segnale, circuito o sistema.
Ciclo di ripristino: Ciclo utilizzato nel funzionamento delle memorie dinamiche per assicurare una memorizzazione affidabile delle informazioni.
Circuito aperto: Percorso ad alta resistenza non aspettato in un circuito.
Circuito integrato (IC): Termine generico dato ai componenti allo stato solido degli attuali sistemi digitali. Gli IC a piccola scala (SSI) generalmente hanno meno di 100 transistori per chip. Quelli a media scala (MSI) ne hanno generalmente più di 100. Quelli di grande scala (LSI) ne hanno generalmente più di 1000.
Clip logica: Spia luminosa che indica il dispositivo utilizzato per controllare i segnali digitali ai piedini di un circuito integrato.
Codificatore: Convertitore di codici che trasforma un codice esterno in un codice del sistema.
Comparatore: Blocco d'accumulo che confronta due numeri binari. Ci sono diversi tipi di confronto: quello per determinare se due numeri sono uguali, quello per determinare quando un numero è maggiore di un altro, quello per determinare quando un numero è maggiore o uguale ad un altro, e così via.
Condensatore: Componente che accumula carica elettrica.
Contatore: Speciale tipo di registro composto da circuiti flip-flop con un ingresso e con, di solito, un'uscita in parallelo per ogni flip-flop, che conta gli impulsi che arrivano all'ingresso e ne memorizza il numero totale secondo un certo codice (solitamente un codice binario).
Contatore di programma: In un sistema programmato sono le parti che forniscono un metodo per incrementare di 1 l'indirizzo dell'istruzione che si sta eseguendo. Può essere formato dal registro indirizzo e da un addizionatore.
Controllo di parità: Controllo che viene eseguito per stabilire se il numero di 1 in una matrice di cifre binarie è dispari o pari.
Controllore (Unità di controllo): Le parti (talvolta un sottosistema) di un sistema programmato che selezionano le istruzioni memorizzate, le interpretano e trasmettono segnali di controllo alle altre parti del sistema.
Convertitore di codice: Nome di una classe di blocchi combinatori di accumulo che ricevono informazioni in un determinato codice e le ritrasmettono secondo un altro codice. Rispetto ad un particolare codice (per esempio il codice BCD utilizzato per i numeri nei calcolatori), un blocco che converte in questo codice viene chiamato "codificatore" mentre un blocco che converte da questo codice viene detto "decodificatore".
CPU: Unità centrale di elaborazione (Central Processor Unit). È la parte principale di qualunque processore o microprocessore. La CPU esegue tutte le funzioni d'elaborazione della ALU, di generazione

d'indirizzi, di temporizzazione e di controllo.

Corto circuito: Un collegamento a bassa resistenza non voluto tra due punti del circuito.

Dati: Informazioni.

Decodificatore: Convertitore di codice che traduce il codice del sistema in un altro codice. Esso "riconosce" cioè una o più combinazioni di bit di ingresso e emette un segnale quando queste combinazioni vengono ricevute.

Diagramma di flusso: Diagramma che mostra il flusso logico o la sequenza del programma di un computer o di un microprocessore.

Digitale: Informazioni costituite da valori o livelli discreti, non continui.

Dispositivi di ingresso/uscita (I/O): Componenti hardware di un computer che servono per introdurre dati in un sistema digitale o per registrare il risultato di un'elaborazione per un uso immediato o futuro.

EPROM: Memoria di sola lettura cancellabile e programmabile (Erasable Programmable Read-Only Memory). Memoria a semiconduttori il cui contenuto può essere cancellato tramite l'esposizione ai raggi ultravioletti, permettendo così al dispositivo di essere riutilizzato per un nuovo programma. La più estesa EPROM attualmente disponibile è di 64K.

Esadecimale: Sistema di misurazione in base 16, comunemente usato nella maggior parte degli odierni computer. I numeri binari vengono uniti in numeri di 4 (per esempio 0101) e hanno valori da 0 a 9 e da A ad F (per numeri da 10 a 15).

Fase: Relazione tra un segnale ed un dato riferimento. I segnali elettrici vengono generalmente considerati come se fossero costituiti da cicli di 360 gradi il cui riferimento è costituito dal segnale di clock. La relazione di fase è generalmente il confronto tra un qualsiasi segnale e il riferimento comune.

Flip-flop R-S: Uno dei vari tipi di flip-flop in cui un 1 momentaneo all'ingresso R (reset) cambia l'uscita "vera" in 0 mentre un 1 momentaneo all'ingresso S (set) cambia l'uscita "vera" in 1. Alcune varietà di flip-flop hanno anche caratteristiche di clock e di "master-slave".

Glitch: Termine dato ad un segnale casuale che provoca errori nei dati o negli indirizzi.

Hardware: La parte fisica (o materiale) che compone un qualsiasi sistema.

Impulsatore logico: Dispositivo utilizzato per inviare impulsi digitali in un sistema.

Indirizzo: Solitamente è un insieme di 8 o 16 bit utilizzato per accedere ad un dispositivo periferico o ad una locazione di memoria.

Ingresso di abilitazione del chip: Ingresso di controllo che, quando è attivo, permette il funzionamento del circuito integrato mentre, quando è inattivo, ne provoca l'inattività.

Interfaccia: Termine dato al circuito che esegue l'adattamento dei segnali di I/O tra il processore e un dispositivo peiferico.

Intermittente: Condizione che si presenta casualmente e con intermittenza in un sistema.

Istruzione: Un passo del programma di un microprocessore.

K: Kilo o mille. Usato generalmente per riferirsi ad un numero composto da un numero intero di migliaia (come per esempio 2K, 4K). Nelle memorie, a causa del sistema binario, K si riferisce ai multipli di 1024 (210).

Linguaggio assemblativo: Termine dato ad un linguaggio di basso livello usato per programmare la CPU.

Logica: Si riferisce al circuito nei sistemi digitali e al sistema operativo su cui è basato il funzionamento del circuito stesso.

Logica combinatoria: Circuiti digitali le cui uscite dipendono solo dagli ingressi presenti.

Logica positiva: Nei circuiti elettrici digitali binari si intende con questo termine la decisione di assumere il "più alto", più positivo fra due livelli di tensione per rappresentare l'1 e di assumere il "più basso", più negativo per rappresentare lo 0. In questo libro si è sempre usata la logica positiva salvo nei casi in cui è stato specificato che si trattasse di "logica negativa".

Logica sequenziale: Uno o più flip-flop o unità di memoria, generalmente con una o più porte logiche e con diversi ingressi e diverse uscite. La combinazione di bit alle uscite non dipende solo dalla combinazione presente in quel momento agli ingressi, ma anche dalla precedente sequenza di combinazioni di ingressi per un certo periodo di tempo.

Memoria: Parte di un computer dove vengono immagazzinati dati o istruzioni finché il processore è pronto per elaborarli.

Memoria dinamica: Memoria di lettura/scrittura dalla quale si può leggere o nella quale si può scrivere periodicamente perché mantenga l'immagazzinamento di informazioni.

Memoria statica: Memoria che mantiene le informazioni immagazzinate finché viene alimentata.

Microcomputer: Termine dato ad un sistema progettato per applicazioni specifiche o ad un sistema che utilizza come sua CPU un microprocessore.

Microprocessore: Nome dato ad un dispositivo LSI o VLSI che contiene tutte le funzioni necessarie ad un processore (CPU, ALU, generazione di indirizzi e segnali di temporizzazione e di controllo).

Mnemonico: Termine dato ad un'abbreviazione che definisce un passo in linguaggio assemblativo nel programma di un microprocessore.

Multiplexer: Vedi "Selettore di dati".

n: Lettera utilizzata quando ci si riferisce a numeri casuali o incogniti.

Nodo: Termine dato ad un punto fisico di connessione in un circuito elettrico.

Oscillatore: Circuito che genera un segnale con una forma d'onda ripetitiva o periodica.

Oscillazione: Termine dato ad un segnale che fluttua sopra e sotto un dato livello di riferimento.

Oscilloscopio: Strumento utilizzato per visualizzare su uno schermo CRT segnali elettrici in funzione del tempo.

Parità: Nome dato ad una serie di bit di controllo di eventuali errori durante la trasmissione dei dati. In caso di parità pari se una certa sequenza di bit dati ha un numero pari di 1, nella sua posizione di bit di parità viene assegnato uno 0. Se invece una sequenza di bit dati ha un numero dispari di 1, nella posizione di bit di parità viene assegnato un 1 in modo che il numero risulti ancora pari.

Periodo: Tempo che intercorre tra due punti corrispondenti di un segnale periodico.

Porta: "Porta logica" che può essere di tipo AND, OR, NOT, NAND, NOR o "OR Esclusivo". In un transistore MOS la "porta"(gate) è la placca metallica utilizzata per mantenere la carica che controlla il transistore.

PROM: Memoria programmabile a sola lettura (Programmable Read Only Memory). Un programma in una PROM viene "stampato" elettricamente e, una volta inserito, non può essere alterato senza utilizzare una nuova PROM.

Processore: La parte di un computer che svolge tutte le funzioni decisionali, aritmetiche, di temporizzazione, di controllo e di invio.

RAM: Memoria ad accesso casuale (Random Access Memory). Memoria con un dato numero di locazioni dove le parole possono essere "scritte" (memorizzate) o "lette" (estratte) in un qualunque ordine casuale.

Registro: Memoria temporanea per dati entranti o uscenti dal processore o per dati non ancora completamente elaborati.

Registro in parallelo: Due o più flip-flop (o unità di memoria dinamiche) che hanno un segnale di clock comune e che vengono utilizzati per memorizzare i bit trasmessi in parallelo.

Rilevazione del segnale: Metodo di ricerca dei guasti col quale si immette in un punto del sistema un segnale noto e lo si controlla in vari punti del sistema per verificarne il percorso.

Ringing: Tendenza di un segnale digitale a superare il livello desiderato. Normalmente si smorza rapidamente lasciando così che il segnale si stabilizzi al suo corretto livello.

ROM: Memoria a sola lettura (Read Only Memory). Si tratta di un dispositivo di memoria che normalmente viene programmato in fabbrica dal costruttore.

Rumore: Qualsiasi segnale non aspettato. Il rumore elettrico può essere causato da piccole e irregolari scintille che possono verificarsi aprendo o chiudendo un interruttore. Esso potrebbe essere provocato da onde radio o da campi elettrici o magnetici generati da un filo e captati da un altro.

Segnale: Parola usata per descrivere il funzionamento di circuiti elettrici o elettronici. Indica la tensione elettrica o la corrente o le onde che trasportano informazioni, oppure l'informazione stessa.

Selettore di dati: Blocco di logico combinatorio che invia i dati provenienti da uno dei diversi ingressi ad una singola uscita, in base ai segnali di controllo. È chiamato anche "multiplexer". Due o più di questi selettori ad un bit operanti in parallelo vengono chiamati "selettore dati a due (tre, quattro, ecc.) bit".

Sincrono: Si riferisce a due o più eventi che vengono fatti accadere contemporaneamente in un sistema per mezzo di un segnale di clock comune.

Sonda logica: Spia luminosa che indica il dispositivo utilizzato per controllare i punti di un segnale digitale ai livelli 1 e 0 in un sistema.

Stato: Lo "stato logico" di un conduttore in un circuito digitale indica la sua condizione in rapporto al fatto che stia trasportando un 1 o uno 0.

Temporizzazione: Termine usato per descrivere il riferimento di clock e/o le sue relazioni con altri segnali.

Terminale CRT: Terminale con tubo a raggi catodici (Cathode Ray Tube). Terminale di un computer con uno schermo simile a quello di un televisore, abbinato ad una tastiera.

Transistori MOS: Classe di transistori che funzionano per mezzo di un campo elettrico prodotto da una tensione su di una placca metallica detta "gate". Il campo agisce attraverso un sottile strato di ossido non conduttore posto sopra un canale semiconduttore, controllando la profondità di quest'ultimo e, di conseguenza, controllando la corrente che può passare attraverso di esso.

Transizione: Termine usato per descrivere il passaggio di un segnale digitale da un livello ad un altro.

Trasmissione di dati in parallelo: Si dice che due o più bit di un gruppo vengono trasmessi "in parallelo" quando essi vengono trasmessi contemporaneamente (come se fossero su più fili) dalla stessa sorgente alla stessa destinazione.

Trasmissione seriale dei dati: Tecnica di trasmissione in cui tutti i bit dati vengono trasmessi in successione (uno dopo l'altro) su di un solo filo.

Indice analitico

Biblioteca Centro Culturale
RAFFAELLO
Via Tuscolana
Circ.ne X Comune di Roma
CIRCOSCRIZIONE ROMA
BIBLIOTECA COMUNALE
621.381/CAN
9250
2178

Analizzatore di firma, 239, 240
Analizzatori logici, 247-248
BCD (Binario codificato decimale), 24-25
Binario codificato decimale, 24-25
Bit
  ingresso a mappa di, 182-183
  uscita a mappa di, 180-182
Buffer a tre stati, 48-50, 97-98
Bus
  concetto, 42
  dati, 42, 142
  di controllo e tempificazione, 42
  indirizzi, 43
Caratteristiche di trasferimento, 11
Circuiti
  in corrente alternata, 5
  in corrente continua, 2
  LSI, 97, 102, 107
  MSI, 97
  SSI, 97
Circuito aperto, 62-64
  in corrente continua, 63
  in un circuito logico, 64, 67
Clip logica, 76
Codice ASCII, 27, 29
Codificatori, 113-115
Contatori, 139-140
Conversione
  analogico-digitale, 30, 32
  da numero binario
    a decimale, 26
    a esadecimale, 26
  digitale-analogica, 32
Convertitore
  A/D, 30, 32
  D/A, 32
Corto circuiti, 59-62
  in un circuito in corrente continua, 59-61
  in un circuito logico, 61-62, 63
CRT, 201-204
Decodificatori, 46-48, 104-105, 239
Distorsione dei segnali, 14
Elaborazione, 35-36
Fase, 6, 8, 209, 211
Flip-flop, 16
  asincrono, 16
  di tipo D, 132
  sincrono, 18
Frequenza, 6, 210
Funzionamento
  del transistore, 66
  ingresso-uscita, 143
Guasti nei componenti, 84-88
Impulsatore logico, 76-77
Ingresso, 35
  a mappa di bit, 182-183
Ingresso/uscita funzionamento (I/O), 143, 179, 213
Invertitore, 16
I/O in parallelo, 183
I/O a mappa di memoria, 185
Istruzioni, 43
Legge di Kirchoff, 4
Legge di Ohm, 2-4
  in tensione alternata, 6
Lettura di memoria, 142
Linguaggio
  assemblativo, 147
  assembly, 147-148
Memoria, 36-37, 50, 224, 233
  a mappa di, 177
  a sola lettura, 53-54
  di lettura-scrittura, 50-51
  EPROM, 54, 174
  funzionamento della, 141-142
  indirizzamento della, 51-53
  lettura della, 142
  PROM, 51
  RAM, 50, 153, 233
  RAM dinamica, 153, 157-165, 233
  ROM, 54, 233
  scrittura in, 142
  sottoinsieme di, 156-157, 160
Microcomputer, 140
Microprocessore, 140
Multimetri, 73-76
Multiplexer
  analogico, 102-104
  digitale, 105-106
Numeri negativi, 27
Oscillatori, 137-139
Oscilloscopio: Appendice, 10, 215, 237
Parità, 127, 190-191
Porta
  AND, 17
  OR, 17
  OR Esclusivo, 17, 236
  XOR, 17, 236
Processori, 54, 143
  generazione di indirizzi, 55
  unità aritmetico-logica, 55-56
  unità di controllo, 54-55
Programma, 144-147
Registri, 41
Rilevatore di glitch, 247
Rilevazione dei segnali, 89-93
Ritardo di propagazione, 11, 212
RS232, 191
Rumore, 14
Scrittura in memoria, 142
Segnali

  analogici, 8
  digitali, 8-9
  distorsione dei, 126
  rilevazione dei, 89-93
- Sistemi
  - combinatori, 38, 161-165
  - sequenziali, 38, 40, 119, 217
- Sonda logica, 76
- Sottosistema di memoria, 156-157, 160
- Stampanti, 204, 206
- Tastiera, 197
- Tempo
  - di discesa, 11, 209
  - di salita, 11, 209
- Temporizzazione, 160, 209-215
  - clock monofase, 209-210
  - clock multifase, 210-211
  - ingresso/uscita problemi di, 222
  - segnali in ritardo, 223-224
- Test
  - basato su computer, 258-259
  - del video, 259-261
  - della tastiera, 261-262
  - delle memorie, 259
- Transistori, 64, 66, 67, 68
  - FET, 70-71
    - a canale N, 71
    - a canale P, 71-72
  - PNP, 70-71
- Trasmissione di dati
  - in parallelo, 20, 183
  - seriale, 20, 189-191
- Uscita, 36
  - a mappa di bit, 180-182
- Verifiche di funzionamento, 168
  - del microprocessore, 251
  - di memoria (RAM), 168, 234
  - di memoria (ROM), 173-174

Risposte alle domande

Capitolo 1

1. c
2. a
3. b
4. b
5. b
6. d
7. d
8. b
9. a
10. a
11. d
12. c
13. c
14. c

Capitolo 2

1. c
2. f
3. d
4. c
5. b
6. b
7. a
8. a
9. c
10. d

Capitolo 3

1. c
2. c
3. a
4. d
5. b
6. a
7. c
8. c
9. d
10. c
11. a
12. c
13. c
14. b
15. c
16. d
17. c
18. c
19. d
20. e

Capitolo 4

1. c
2. b
3. a
4. c
5. c
6. c
7. c
8. a
9. b
10. c

Capitolo 5

1. b
2. b
3. a
4. b
5. c
6. a
7. a
8. d
9. b
10. d
11. c
12. c

Capitolo 6

1. b
2. c
3. c
4. a
5. b
6. d
7. d
8. b
9. e
10. b
11. b
12. c
13. b
14. e
15. c
16. a
17. d
18. a
19. a
20. a

Capitolo 7

1. c
2. b
3. d
4. e
5. d
6. a
7. d
8. a
9. b
10. e

Capitolo 8

1. a
2. c
3. b
4. e
5. c
6. a
7. b
8. c
9. c
10. d

Capitolo 9

1. c
2. d
3. d
4. a
5. c
6. a
7. c
8. a
9. b
10. c
11. f
12. a
13. d
14. f
15. c
16. d
17. e
18. a
19. a
20. c

Collana Capire l'Elettronica e l'Informatica

Capire • l'Elettronica Digitale: Guasti e Manutenzione

Vi spiega le tecniche fondamentali di riparazione dei guasti in modo facile da capire e da apprendere

di Don L. Cannon

tecniche nuove

IL LIBRO

L'elettronica digitale continua ad assorbire un numero sempre crescente di funzioni tipiche di sistemi elettrici, meccanici ed elettromeccanici. I sistemi elettronici digitali sono più affidabili dei sistemi che sostituiscono, ma viene un momento che anche essi hanno bisogno di riparazione e di manutenzione. Questo volume fornisce una visione nel mondo di queste tecniche, usando un linguaggio che sia i tecnici che i non-ingegneri possono facilmente capire. Il volume presenta i concetti di base e le tecniche fondamentali necessari per localizzare i diversi tipi di guasti nei sistemi digitali e per provvedere alla loro riparazione.

CONTENUTO DEL VOLUME

Partendo dai concetti fondamentali dei sistemi digitali e procedendo attraverso le tecniche di temporizzazione dei sistemi più avanzati, la riparazione dei guasti è descritta in 9 capitoli:

1. Fondamenti di elettronica digitale

Una panoramica sui sistemi digitali, dalla legge di Ohm alla conversione binario-decimale. Parla di funzionamento seriale, parallelo, sincrono e asincrono.

2. Funzioni di sistemi digitali

Che cosa è un sistema combinatorio o un sistema sequenziale? Cosa sono i registri, i buffer 3-state, le ROM, le RAM, le ALU, le CPU? Cosa è un'istruzione, un bus?

3. Fondamenti di riparazione dei guasti

Una panoramica dei problemi più comuni in un sistema digitale, tecniche di misura e modi di ricerca del guasto. Iniziando dai cortocircuiti e dai circuiti aperti, conclude con le tecniche di tracciamento del segnale. Spiega l'uso dei clip logici e delle sonde logiche.

4. Problemi di logica combinatoria

Spiega come individuare problemi tipici di malfunzionamento in circuiti di logica combinatoria SSI e MSI, come multiplexer, codificatori, decodificatori, memorie a sola lettura.

5. Problemi di logica sequenziale

Spiega come individuare problemi tipici di malfunzionamento in circuiti di logica sequenziale. Tra gli esempi è compreso un sistema di controllo a sei stati che viene sostituito con un microcalcolatore.

6. Problemi connessi con le memorie

Capire i diversi tipi di memorie e le procedure necessarie per testarle e individuarne i guasti.

7. Problemi di ingresso/uscita (Input/Output)

Illustra gli ingressi e le uscite a mappa di bit, la trasmissione dati seriale, lo standard RS232, il funzionamento di una tastiera e di uno schermo CRT.

8. Problemi di temporizzazione

Uno sguardo a come i circuiti sono sincronizzati dagli impulsi di temporizzazione del generatore del segnale di clock principale. Uso dell'oscilloscopio per osservare le forme d'onda di un sistema sincrono.

9. Tecniche avanzate

Spiega l'uso dell'analizzatore di stati logici e dei sistemi a microprocessore di test automatico per individuare i guasti in sistemi digitali di alta complessità e di elevate prestazioni.

COLLANA Capire l'Elettronica e l'Informatica

I volumi della collana Capire l'Elettronica e l'Informatica, a cura del "Learning Center" della Texas Instruments, costituiscono una raccolta scritta per **chiunque** voglia in modo semplice e rapido conoscere di più dell'elettronica moderna e dell'era del personal computer, del loro impatto nel nostro mondo e delle loro applicazioni nella nostra vita. Ogni volume è scritto con un linguaggio chiaro, semplice e vicino alla realtà, e approfondisce un aspetto specifico di
ciò che vi è di più recente
nell'elettronica moderna. Si
siate un hobbista, un amat
tecnico, coinvolti con l'elett
l'informatica, o sia che siat
semplicemente interessati p
curiosità a cosa sta accade
mondo dell'elettronica, delle
applicazioni, dei calcolatori
nuovi microprocessori, sicu
i volumi della collana trover
che risponde ai vostri intere

L. 39500

ISBN 88 7081 317 7

Capi

Biblioteca
Comunale
L1
621.
381
CAN

ica Digitale: Guasti e Manutenzione Don L. Cannon